

Uma Nova Arquitetura Descentralizada para Mobilidade de Fluxo em Redes Veiculares

Douglas D. Lieira¹, Edivaldo P. Valentini^{1,2}, Luis H. V. Nakamura¹,
Rodolfo I. Meneguette¹

¹ Instituto Federal de São Paulo - IFSP

² Universidade Estadual de São Paulo - UNESP

{ddlieira, edivaldo, nakamura, meneguette}@ifsp.edu.br

Abstract. *Intelligent transportation system services aims to optimize the monitoring and control of a city's traffic. These services are usually run in clouds or in local datacenters where they have abstracted and processed information from the vehicle to assist the transport system. Therefore, it is necessary that the vehicle needs to be connected to the internet so that it can have access to these services, that is, the vehicle will have to connect with a road infrastructure. In order for the vehicle not to have its service interrupted due to an access point exchange, an efficient mobility mechanism is necessary. Thus, in this article, we propose a decentralized architecture in which the shoreline infrastructure is organized to manage the mobility the vehicle between several connections along its trajectory. The proposed architecture seeks to meet the quality of service parameters of each application without overloading the network. For the evaluation of the proposed architecture as well as to make a comparison with three works in the literature we use NS3 simulator. The results showed that the proposed mechanism decreases packet loss (approximately 8%), maximizes input of information (90%), and decreases handover time (0.2 s).*

Resumo. *Serviços de sistema de transporte inteligentes visam otimizar o monitoramento e o controle do tráfego de uma cidade. Geralmente esses serviços são executado em nuvens ou em datacenters locais os quais realizaram a abstração e o processamento das informações vindas do veículo para auxiliar o sistema de transporte. Portanto, é necessário que o veículo esteja conectado à Internet para que possa ter acesso a esses serviços, ou seja, o veículo terá de se conectar com uma infraestrutura de acostamento. Para que o veículo não tenha seu serviço interrompido devido a uma troca de ponto de acesso é necessário de um mecanismo de mobilidade eficiente. Assim, neste artigo, propomos uma arquitetura descentralizada na qual a infraestrutura de acostamento se organizem para gerenciar a mobilidade o veículo entre diversas conexão ao longo de sua trajetória. A arquitetura proposta busca atender os parâmetros de qualidade de serviço de cada aplicação sem sobrecarregar a rede. Para a avaliação da arquitetura proposta bem como realizar uma comparação com três trabalhos na literatura utilizamos simulador NS3. Os resultados mostraram que o mecanismo proposto diminui a perda de pacotes (aproximadamente 8%), maximiza a entrada das informações (90%) e diminui o tempo de handover (0,2 s).*

1. Introdução

O aumento dos serviços e aplicações voltados para a otimização de tarefas do nosso cotidiano vem cada vez mais ganhando espaço, não só nas empresas, mas também junto

aos cidadãos de uma grande cidade [Meneguette 2016, Chhabra et al. 2017]. Um exemplo disso é a crescente quantidade de novos serviços voltados aos sistemas de transporte inteligentes, que buscam trazer uma otimização em todo sistema de transporte, provendo informações das condições de tráfego, serviços de páginas amarelas, entre outros. Esses sistemas de transporte utilizam das redes veiculares e de suas aplicações como ferramenta de auxílio para abstração de informações e disponibilização de recursos e serviços para os seus usuários [Ramazani and Vahdat-Nejad 2014, Lima et al. 2017].

As aplicações voltadas para o sistema de transporte inteligente podem ser [Meneguette et al. 2013b, Jazar 2017] classificadas em serviços de seguranças que tem como objetivo disponibilizar para o motorista informações relacionadas com o tráfego de carros em seu percurso, engarrafamentos, condições da estrada, acidentes entre outras. Serviços de conforto que fornecem informações de estabelecimentos próximos ao condutor, tais como lugares para realizar o abastecimento do veículo, restaurantes próximos, entre outros serviços. E por fim, serviços de entretenimento que visam o compartilhamento e a propagação de conteúdo voltado ao entretenimento de seus usuários. Grande parte dessas aplicações são disponibilizadas pela Internet através de serviços de *clouds* e *data-centers*. Portanto, para que um condutor possa ter acesso a alguns serviços dos sistemas de transporte inteligente, o veículo precisará estar conectado com algum ponto de acesso para ter acesso a Internet.

Devido às características singulares das redes veiculares tal como a alta mobilidade do veículo, a troca de um ponto de acesso ao outro, o veículo poderá ter seu fluxo de informação interrompida devido à troca de ponto de acesso [Sawarkar and Madankar 2016]. Para diminuir essa perda de conectividade e evitar que uma determinada tecnologia de rede fique sobrecarregada, é necessário um mecanismo de mobilidade de fluxo [Balfaqih et al. 2017]. Esses mecanismos fornecerão uma troca entre pontos de acesso de forma transparente e sem descontinuidade do fluxo de informações de uma determinada aplicação ou serviço. Além do gerenciamento de mobilidade alguns desses mecanismos também oferecem um balanceamento de carga entre as interfaces de rede, ou seja, esses mecanismos realizarão o gerenciamento de múltiplas interfaces de rede para que possam ser utilizadas simultaneamente [Su et al. 2017]. Esse gerenciamento permitirá o mapeamento entre as interfaces de rede ativas no dispositivo com os fluxos de informações geradas pelas aplicações disponibilizadas na Internet.

Na literatura existem diversas arquiteturas que lidam como o gerenciamento de mobilidade na qual considera o uso de múltiplas interfaces de rede simultaneamente [Makaya et al. 2012, Lin et al. 2017, Chen-wen and Ping 2009, Meneguette et al. 2013a, Meneguette et al. 2016, Su et al. 2017], considerando tanto uma mobilidade entre tecnologias de rede diferentes, *handover* vertical, quanto para a mesma tecnologia, *handover* horizontal. Essas soluções utilizam um elemento centralizador para realizar o gerenciamento das interfaces, bem como dos fluxos de informações que a rede está trafegando. Por serem centralizadas, essas arquiteturas possuem um elemento único de falha, ou seja, se o elemento controlador vier a falhar o sistema poderá parar de funcionar, além disso essas soluções precisam de diversos elementos para fornecer uma escalabilidade para a infraestrutura de serviço.

Dessa forma, é necessário uma arquitetura de comunicação descentralizada e flexível que permita o gerenciamento de mobilidade das múltiplas interfaces de rede ativas

simultaneamente, considerando tanto pontos de acesso com a mesma tecnologia de redes, como tecnologias diferentes. Portanto, esse trabalho tem como objetivo desenvolver uma arquitetura descentralizada para o gerenciamento da mobilidade de fluxos das aplicações utilizadas de forma que esse gerenciamento seja transparente para o usuário. Para isso consideramos que a RSU (*RouteSide Unit*) tem um poder computacional capaz de armazenar tabelas de roteamento e encaminhamento. Assim as RSUs ficarão responsáveis pelo gerenciamento do fluxo de informações e da mobilidade do veículo. A RSU proverá uma descentralização de recursos e serviços, permitindo um gerenciamento de mobilidade mais próxima aos veículos, permitindo uma comunicação heterogênea com diversas tecnologias de rede, tentando atender não somente as necessidades de cada tipo de rede, mas também atender os requisitos de qualidade de serviço de cada tipo de serviço disponibilizado na Internet. Assim, a contribuição desse trabalho é o desenvolvimento de uma arquitetura baseada no controle da RSU na qual gerenciará as múltiplas interfaces de rede ativas no ambiente de modo a atender as necessidades de cada classe de aplicação. Além disso, fornecer ferramentas de um gerenciamento distribuído da mobilidade do fluxo de informações na rede.

O restante do trabalho está estruturado da seguinte maneira. A seção 2 apresenta os trabalhos da literatura. A seção 3 propõe um mecanismo descentralizado para o gerenciamento de mobilidade de fluxo em redes veiculares. Os resultados da comparação do mecanismo proposto em comparação com outras soluções da literatura são apresentados na seção 4. A seção 5 apresenta a conclusão e os trabalhos futuros.

2. Trabalhos Relacionados

Esta seção apresenta alguns trabalhos relacionados que focam no mecanismo de mobilidade de fluxo, buscando realizar uma troca de ponto de acesso de forma transparente aos usuários, sem que haja uma descontinuidade do fluxo de informações entre a central de serviços (*cloud e/ou datacenter*) e os veículos. Esses trabalhos também permitem o uso simultâneo de múltiplas interfaces durante a mobilidade do veículo para assegurar o fluxo contínuo de dados.

SIPTO (*selective IP traffic offload*) [Makaya et al. 2012] proposto por Makaya et. al. é um mecanismo que provê um suporte para *offload* de dados para dispositivos móveis equipados com mais de uma interface de rede. Tal mecanismo considera o estado da rede como parâmetros de decisão, ou seja, se o dispositivo deverá ou não realizar a mudança de fluxo de uma interface para a outra. Para a abstração do estado da rede esse mecanismo utiliza o protocolo 802.21, na qual traz um conjunto de pacotes padrão para o gerenciamento de mobilidade em redes sem fio. Tal protocolo traz consigo um conjunto de mensagem de controle que aumenta o *overhead* na rede prejudicando o seu desempenho e aumentando a possibilidade de colisões e perda de informações na rede.

IF-HMIPv6 (*improved fast handover scheme for hierarchical mobile IPv6*) [Chen-wen and Ping 2009] foi desenvolvido por Chen-wen e Ping [Chen-wen and Ping 2009] com objetivo de reduzir o atraso de transferência e perda de pacote. Para isso o Mobile Node (MN) envia a mensagens de solicitação de roteador para o *proxy* - mensagem (IRtSolPr). A mensagem IRtSolPr contém informações sobre o endereço MAC ou o identificador do AP. Além disso, como uma maneira de identificar o MAP que MN segue durante o processo de IF-HMIPv6, um "I Flag" foi incluído na mensagem IRtSolPr. Ao utilizar este "I Flag", MAP foi capaz de processar o

novo processo de configuração CoA (*Care-of Address*) e envia uma mensagem *handover initiation* (HI) para o NAR (*new Access Router*), a fim de estabelecer um túnel.

SFMMA (Seamless Flow Mobility Management Architecture) [Meneguette et al. 2013a] foi proposta por Meneguette et. al. na qual consiste de uma arquitetura que provê multi-acesso de forma transparente em redes sem fio. Essa arquitetura tem como objetivo oferecer uma conexão contínua e transparente para as aplicações veiculares, mantendo os requisitos mínimos das aplicações veiculares, como por exemplo: atraso, perda de pacote, bem como maximizar o tráfego de pacotes na rede. Para isso, a arquitetura possui um módulo de gerenciamento de fluxo, o qual é baseado nas classes de aplicação das redes veiculares. Esse módulo também considera o estado de cada rede ativa no ambiente na qual o veículo está trafegando. SFMMA possui um alto *overhead* devido aos protocolos utilizados para abstração do estado de rede e controle do *handover*.

MAVEN (flow mobility MAnagement for VEhicle Networks) [Meneguette et al. 2016] proposto por Meneguette et. al. é uma evolução do SFMMA na qual os autores modificaram o protocolo 802.21 visando diminuir a quantidade de mensagens transmitidas para a abstração do estado da rede e gerenciamento de mobilidade da arquitetura. Além disso, MAVEN removeu os mecanismos de decisão de mudança tanto no *gateway* quanto no Nó móvel, assim a decisão de qual rede irá mudar fica somente a cargo do controlador, diminuindo assim trocas desnecessárias. Embora as modificações tenham melhorado o desempenho do mecanismo de gerenciamento de mobilidade, a arquitetura ainda apresenta um único ponto de falha, ou seja, se o controlador da arquitetura falhar toda a gerência e o roteamento dos fluxos de dados também vão parar.

Neste trabalho é proposto uma nova arquitetura que busca maior flexibilidade e escalabilidade no gerenciamento de mobilidade através do uso da capacidade de armazenamento e roteamento das RSUs. Diferentemente das demais arquiteturas que são centralizadas e possuem um ponto único de falha, a solução proposta tem como objetivo a distribuição do controle entre os veículos, bem como utiliza as RSUs como um mecanismo de auxílio ao controle dos parâmetros de rede para que os requisitos mínimos de cada classe de aplicação da rede possa ser atendida sem sobrecarregar nenhuma tecnologia de rede, além de prover maior escalabilidade para o sistema.

3. Uma Nova Arquitetura Descentralizada para Mobilidade de Fluxo em Redes Veiculares

Nessa seção descreveremos uma arquitetura descentralizada para o gerenciamento de mobilidade dos veículos utilizando a capacidade de armazenamento e roteamento das RSUs. DELMARVA arquitetura DEscentraLizada para MobilidAde de fluxo em Redes VeiculAres, a qual consiste de uma infraestrutura distribuída para acesso a diferentes tecnologias de redes simultaneamente, de forma transparente e contínua para os serviços e aplicações disponibilizadas na Internet. A arquitetura tem como objetivo maximizar o uso efetivo da rede, assim atendendo aos requisitos de cada classe de serviço na rede, tal como: vazão, atraso, entre outros parâmetros. Para isso, utilizaremos as RSUs para fornecer um mecanismo de gerência de mobilidade de fluxo, próximos aos veículos, capazes de abstrair informações tanto das redes cabeadas como das redes veiculares e permitindo assim

um gerenciamento mais dinâmico não só da comunicação mais também dos fluxos de informação que estão trafegando na rede.

Portanto, a arquitetura proposta considera que a RSU possui capacidade de armazenamento e roteamento na qual um grupo de RSUs será responsável por gerenciar uma região do mapa, assim, o mapa está dividido em regiões que tem o tamanho da área de cobertura da comunicação de uma RSU. Cada RSU opera de forma independente uma das outras, entretanto, devido a tecnologia de rede, haverá sobreposição da área de cobertura, ou seja, o mapa da cidade estará coberto com pontos de acessos fixos que oferecem conectividade tanto wireless (802.11p) quanto pelas redes celulares (LTE). Portanto, a RSU que gerencia a rede de celular poderá sobrepor a RSU que fornece uma conectividade a uma rede wireless.

A RSU proverá o acesso à Internet para os veículos, na qual permitirá a captação de informação do estado de rede, como por exemplo, informações de vazão e perda de pacotes. Além disso, permitirá a abstração das informações sobre os serviços ou fluxo de informações que estão sendo transmitidos em sua região. Para ampliar o conhecimento da RSU, a DELMARVA utiliza-se de um mecanismo de *publish-subscribe* que permitem que RSUs próximas ou sobrepostas possam trocar informações e assim, com um maior conjunto de informações, seja possível tomar uma ação sobre um determinado fluxo de informação.

3.1. Abstração e Processamento das Informações de Controle

A RSU coleta informações através das trocas de mensagens de controle com os veículos que estão próximas ou conectados à ela. Uma dessas mensagens de controle enviadas na rede é o *beacon* que consiste da velocidade do veículo, sua direção, a rota que será percorrida pelo veículo, bem como o tempo que o veículo demorou para percorrer as ruas. Essas informações permitirão que a RSU conheça quais são os veículos que estão em sua região e que, possivelmente poderão ser gerenciados por ele. O *beacon*, além de trazer informações referentes ao veículo, permite que a RSU calcule uma média de quanto tempo o veículo permanecerá em sua região.

Além da comunicação entre veículo e RSU, as RSUs também trocam informações entre elas para construir uma base do conhecimento com não somente das regiões próximas a eles mais também saber as condições das redes que a RSU gerencia. Para isso, as RSUs utilizam o método de *public-subscribe* para propagar informações de mapa, bem como informações de rede. As informações de mapa trocadas entre elas consistem na indicação das ruas que a RSU gerencia, quantidade de veículos na região e tempo médio que o veículo permanece na região. Parte dessas informações são utilizadas pelo processo de decisão de mudança de fluxo. As informações de rede consiste da condição de cada fluxo gerida por ele, ou seja, o a identificação do fluxo, vazão desse fluxo, latência, perda de pacote e a identificação do veículo que está enviando e recebendo informações por esse fluxo. Além dessas informações de fluxo, também são enviadas características da própria rede gerida pela RSU que consiste de informação de vazão total, perda de pacote, quantidade de fluxo, e atraso total da rede.

O algoritmo 1 descreve o recebimento das mensagens do *beacon* e de informações trocadas entre as RSUs. Todas essas informações são armazenadas em três principais tabelas: (I) veículos na região que consiste do armazenamento das informações referentes aos veículos que estão trafegando nas ruas pertencente a RSU, ou seja, o armazena-

mento de todas a informações descritas anteriormente. (II) status das redes sobrepostas que consiste do armazenamento das informações tanto de rede quanto de veículos que estão conectados a uma RSU sobreposta ou próxima a ele, ou seja, armazenamento das informações descritas acima tomando como índice da tabela a identificação da RSU. (III) fluxos das informações que a RSU está gerenciando, ou seja, os fluxos que foram alocadas para a RSU realizar o encaminhamento e o roteamento dos dados. Essa tabela guarda informações das condições de rede de cada fluxo, a perda de pacote, a vazão e a latência. Essa tabela também armazena informações de roteamento que consistem: no IP do veículo, no prefixo da rede a ser roteada e no identificador do fluxo de pacotes.

Algorithm 1 Recebimento das Mensagens Beacon e comunicação entre as RSUs

```

1:  $Message \leftarrow lermensagem()$ 
2: if ( $Message == Beacom$ ) then
3:   Atualizar_Tabela(mensagem)
4: end if
5: if ( $Message == RSUMsg$ ) then
6:   Atualizar_Tabela(mensagem)
7: end if

```

Parte das informações armazenadas nas tabelas são utilizadas pelo mecanismo de decisão de troca de fluxo. Esse mecanismo de controle de fluxo verifica se os parâmetros de rede de um determinado fluxo estão fora dos padrões de qualidade de serviço da aplicações ou se o veículo está saindo da região. Em caso afirmativo, esse mecanismo verifica qual outra RSU poderia gerenciar esse fluxo ou veículo. O mecanismo de seleção de possível RSUs foi utilizado no trabalho de Silva et. al. [Silva et al. 2014] no qual utilizam as técnicas de *Elitist Selection Strategy* [Engelbrecht 2006] e *The Multiple Attribute Decision Making* [Zekri et al. 2012]. Entretanto, como parâmetro de análise, nós utilizamos o estado da rede de cada RSU sobreposta bem como o tempo de permanência de cada região sobreposta. A saída desse mecanismo será a RSU que melhor atende as necessidades do fluxo de dado considerando a mobilidade do veículo pelas diferentes regiões. O algoritmo 2 descreve a realização de uma mudança de fluxo devido as mudanças dos parâmetros de rede do fluxos já alocados.

Algorithm 2 Gerenciamento de mundaça de fluxo pela RSU

```

1: if ( $Estado\_fluxo \leq Parametro\_fluxo$ ) then
2:    $Tempo\_Estinado\_Região \leftarrow getMedTempo(Veículo\_Região)$ 
3:    $Estado\_fluxo \leftarrow getEstadoFluxo(Gerencia\_RSU)$ 
4:    $NovoRSU \leftarrow EscolhaRSU(Tempo\_Permanencia\_Região, Estado\_fluxo)$ 
5:   enviar_para_RSU(troca_RSU)
6:   enviar_para_Veiculo(troca_Fluxo)
7: end if

```

Uma vez decidido a troca, a RSU original envia uma mensagem de controle chamada de "troca_RSU", composta por: ID do fluxo, ID do veículo, o IP do veículo e o prefixo de rede a ser roteado para a nova RSU. A nova RSU ao receber essa mensagem atualiza sua tabela de fluxos (linhas 1-3 do algoritmo 3. Além disso, a RSU original

também envia uma mensagem para o veículo informando sobre a mudança de ponto de acesso para que o veículo comece a encaminhar suas informações para a nova RSU. Essa mensagem entre a RSU e o veículo é denominada "troca_fluxo" na qual possui o ID da RSU, seu endereço e sua localização. Ambas as mensagens de controle possuem uma mensagem de confirmação que informa o status da solicitação, por exemplo, a mensagem de confirmação iria com um status "OK", caso a troca fosse estabelecida. O algoritmo 3 mostra a troca de mensagens feita pelas RSUs para a mudança de fluxo.

Algorithm 3 Gerenciamento de troca de RSU

```
1: if (Message == Troca_RSU) then
2:   Atualizar_Tabela(mensagem)
3:   enviar_para_RSU_estado(ACK)
4: end if
5: if (Message == Nova_Interface) then
6:   Tempo_Permanencia_Região ← getMedTempo(Veículo_Região)
7:   Estado_Rede ← LerEstadoRede(tabela) + LerEstadoFluxo(tabela)
8:   NovoRSU ← EscolhaRSU(Tempo_Permanencia_Região, Estado_fluxo)
9: end if
10: if (RSU == meuid) then
11:   enviar_para_Veiculo(troca_fluxo)
12: else
13:   enviar_para_RSU(troca_RSU)
14:   enviar_para_Veiculo(troca_Fluxo)
15: end if
```

3.2. Procedimento de troca de fluxo

A geração de um fluxo ocorre quando um determinado veículo necessita de um serviço específico na Internet. Para isso, consideramos que o veículo possui pelos menos 3 interfaces de rede, sendo uma interface lógica e duas interfaces físicas. A interface lógica é utilizada para que nas camadas acima não haja uma descontinuidade no envio e recebimento de informações, ou seja, essa interface de rede possuirá um IP permanente. As interfaces físicas correspondem as interfaces de rede na qual se conectarão as RSUs, essas interfaces também possuirão IPs previamente alocados. Assim, o veículo possuirá um módulo para o mapeamento do IP permanente (interface lógica) mais a porta do aplicativo ou serviço com o IP da rede. Assim, quando houver qualquer mudança de seus fluxos o veículo modificará esse mapeamento.

O processo de gerenciamento de mobilidade se inicia quando o veículo precisa enviar ou receber informações da Internet. Então, ele ativa as interfaces de redes e o veículo enviará um mensagem de conexão (mensagem padrão do protocolo sem fio) para a RSU mais próxima. As RSUs ao receberem essa mensagem verificarão qual será a melhor RSU para atender a demanda do fluxo e enviará a mensagem "troca_fluxo" para informar a interface em qual RSU o fluxo será gerenciado, conforme exibido na Figura 1.

Uma vez alocado o fluxo em uma das RSU a troca desse fluxo poderá ocorrer perante dois eventos: (i) Se um parâmetro de um determinado fluxo está fora dos limites esperados (vazão, atraso e a perda de pacote); (ii) o veículo está saindo da região

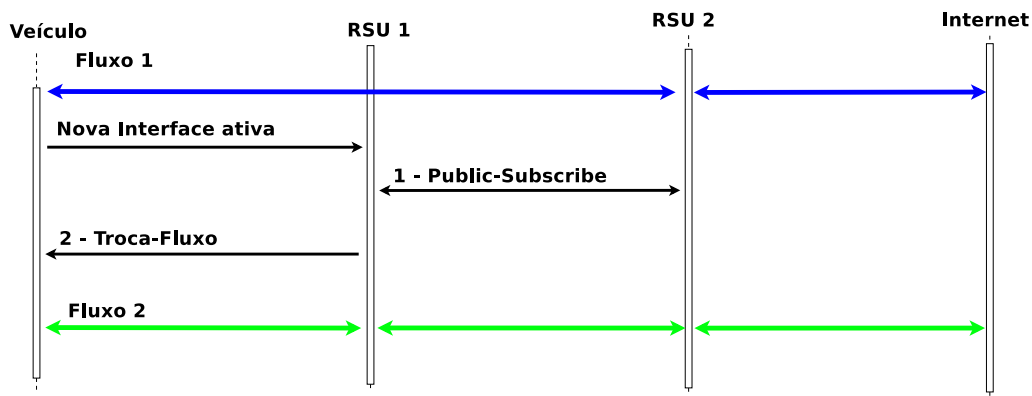


Figure 1. Abstração da inicialização da Interface

gerenciada pela RSU. Para ambos os eventos as RSUs irão chamar o método de seleção para verificar qual é a nova RSU que atenderá esses fluxos uma vez selecionada a RSU e enviado uma mensagem de mudança para a nova RSU que, por sua vez, começará a encaminhar informações desse novo fluxo. Além disso, o veículo também é notificado dessa mudança para que ele possa modificar o mapeamento das interfaces, conforme exibido na Figura 2.

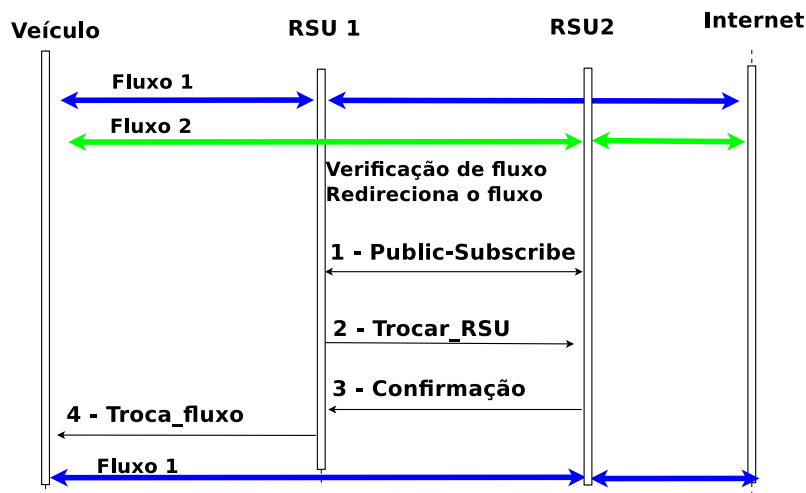


Figure 2. Abstração da troca de fluxo pela RSU

4. Ambiente de Simulação e Análise de Performance

Nesta seção estão descritos em detalhes o ambiente de simulação e a análise do desempenho da solução proposta que foi realizada por meio de experimentos de simulação de redes que fornece os módulos dos protocolos 802.11p, bem como o protocolo LTE para redes celulares.

Além disso, usamos o *Luxembourg SUMO Traffic* (LuST) [Codeca et al. 2015] para capturar a mobilidade de veículos em estradas públicas. LuST é um cenário realista de 24 horas da cidade do Luxemburgo com uma área de $155,95 \text{ km}^2$, 5,969 km de bordas e um total de 295,979 veículos inseridos no cenário (Figura 3). Nesse artigo utilizamos o

primeiro intervalo de tempo (das 00:00 às 06:00). Além disso, nós variamos de 10 % a 50 % os veículos que estarão transmitindo e recebendo informações da Internet, ou seja, apenas uma porcentagem dos veículos receberá e enviará dados para a Internet, outros veículos serão usados apenas para a retransmissão de informações, se necessário. Assim, avaliamos a solução proposta sob diferentes demandas de aplicação. A escolha desses veículos ocorre aleatoriamente para que as soluções sejam tão realistas quanto possível.

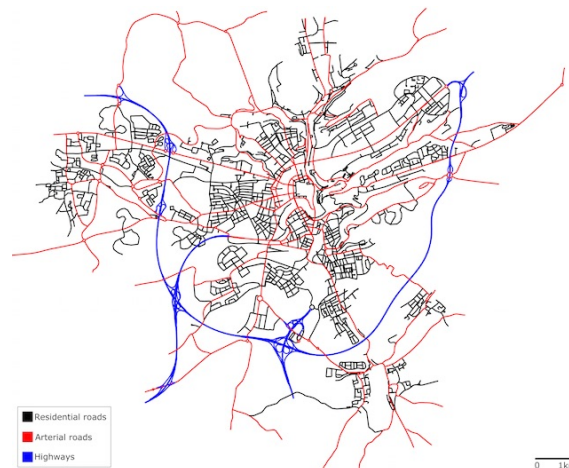


Figure 3. Simulation maps used for the comparison analyses. (Luxemburgo)

Cada veículo selecionado executa três aplicações, sendo que cada uma pertence a uma classe de aplicativos, sendo elas: uma aplicação de entretenimento, uma de conforto e uma de segurança. A frequência de envio e recebimento de informações dessas classes de aplicativos é 1s para aplicativos de entretenimento, 0.5s para aplicações de conforto e 0.1s para aplicativos de segurança. Estes valores foram retirados dos padrões do Instituto Europeu de Normas de Telecomunicações (ETSI) [European Telecommunications Standards Institute 2009]. Portanto, os veículos selecionados usaram uma aplicação de cada classe de aplicativos ao mesmo tempo.

Além disso, todos os veículos possuem duas interfaces de rede, LTE e 802.11p, e ambas as interfaces têm seus endereços previamente atribuídos. Para a configuração do LTE, usamos a configuração padrão do NS-3, que fornece uma área de cobertura de aproximadamente 5 km para o LTE e cobre praticamente boa parte do mapa. No entanto, para as configurações 802.11p usamos o *two-ray ground propagation model* e o raio de transmissão é de 300 m. A tabela 1 descreve os parâmetros da simulação. Portanto, todos os veículos estarão dentro do alcance de um ponto de acesso LTE. Os pontos de acesso sem fio foram espalhados pelo mapa, mas eles não cobrem todos os pontos no mapa.

O objetivo das simulações foi verificar o impacto da arquitetura proposta sobre o fluxo de informações transmitidas na rede. Portanto, pretendemos verificar se a descentralização da gestão de mobilidade do veículo afetou no tempo de mudança de fluxo de uma interface para outra. Para isso, utilizamos quatro métricas para avaliar nossa arquitetura: tempo de entrega, atraso, porcentagem de perda de pacotes e porcentagem de pacote entregues que considera a taxa entre os pacotes de entregues com sucesso e os pacotes enviados. Além disso, para cada cenário de simulação executamos 35 vezes e

Table 1. Parâmetros da Simulação

Parâmetros	Valor
Potencia de Transmissão	2.2mW
Área de transmissão	300m
Frequência da classe de segurança	0.1s
Frequência da classe de entretenimento	1s
Frequência da classe de conforto	0.5s
Quantidade de vezes executadas	35
Intervalo de Confiança	95%

calculamos intervalos de confiança considerando uma taxa de 95%. Comparamos nossa arquitetura com MAVEN [Meneguetto et al. 2016], SFMMA [Meneguetto et al. 2013a], IF-HMIPv6 [Chen-wen and Ping 2009].

Figura 4 mostra o tempo médio de *handover*. Neste gráfico podemos observar que o DELMARVA obteve um tempo de entrega mais curto, devido ao processo de decisão durante a mudança do ponto de acesso pelo veículo. Em outras palavras, a solução proposta reduziu o número de trocas realizadas por veículos, pois a troca só ocorre quando os parâmetros da rede não atendem ao fluxo ou ao veículo que altera o ponto de acesso. Assim, a rota do fluxo entre as RSUs fica transparente para o veículo. Analisando o gráfico, a arquitetura DELMARVA forneceu uma redução de aproximadamente 1,5% em relação ao MAVEN e uma redução de 39% em relação ao IF-HMIP. Essa redução ocorre devido ao conhecimento prévio sobre as condições da rede e seus fluxos, evitando mudanças desnecessárias, além de diminuir a latência da rede, como exibido na Figura 5.

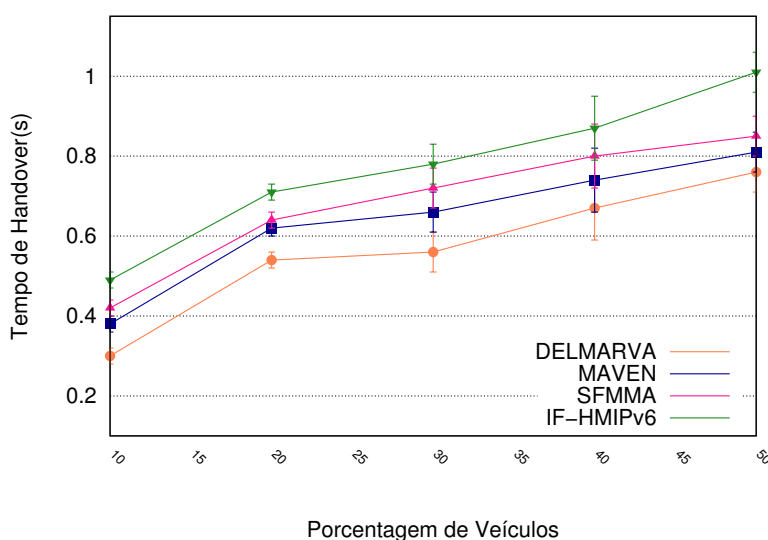
**Figure 4. Tempo Médio de Handover**

Figura 5 mostra o atraso médio de todas as classes de aplicativos. Podemos observar que o DELMARVA tem uma redução média de atraso de aproximadamente 20% em comparação com MAVEN e 40% comparado com o IF-HMIPv6. Quando analisamos

o gráfico com 50% carros usando as aplicações, a solução proposta tem uma redução de cerca de 15% em comparação com MAVEN e 57% em relação ao SFMMA. Isso ocorre porque o DELMARVA fornece um tempo de entrega mais baixo, bem como uma pequena quantidade de pacotes perdidos (Figura 6) que permitem uma redução do valor na retransmissão de pacotes. Além disso, o DELMARVA executa um balanceamento de carga de pacotes entre as várias interfaces ativas no ambiente.

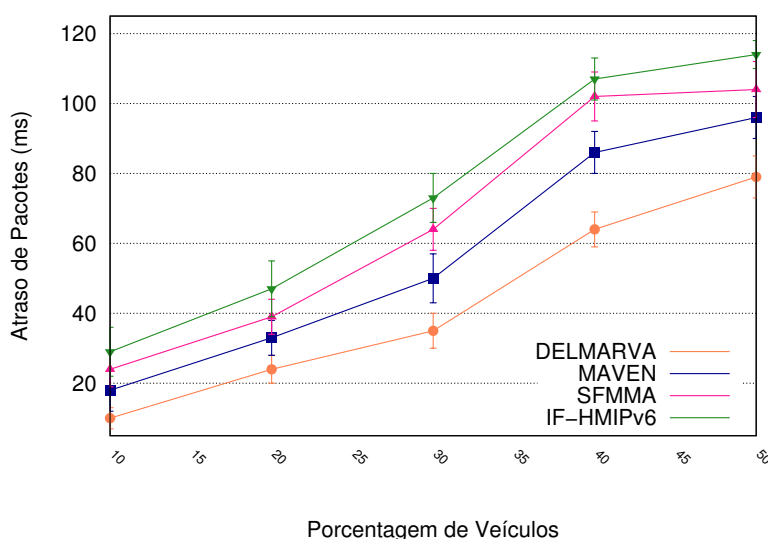


Figure 5. Atraso Médio de Todas as Classes de Aplicativos.

Figura 6 descreve a perda de pacotes. O gráfico demonstra que o DELMARVA teve uma porcentagem menor de perda de pacotes. Como o mecanismo de decisão de troca de entre as RSUs permite um balanceamento de carga para cada interface de rede no veículo, levando em consideração as necessidades de cada classe de aplicação, além de evitar uma sobrecarga e uma certa tecnologia de rede. Portanto, DELMARVA proporcionou uma diminuição na contenção entre os fluxos para obter a interface, bem como uma redução no tempo de transferência. Ao analisar os gráficos, podemos observar que o DELMARVA prova uma redução de 3% na porcentagem de perda dos pacotes em comparação com MAVEN e aproximadamente 30% em comparação com IF-HMIPv6. Esta pequena quantidade de pacote impactou o resultado da porcentagem de pacotes entregues.

Figura 7 mostra a porcentagem de pacotes entregues com sucesso. Podemos ver que a solução proposta DELMARVA obteve uma maior porcentagem de entrega de pacotes sem exigir retransmissões, um desempenho significativo do mecanismo de transmissão em que demonstra uma redução não só na perda de pacotes, mas também uma redução no número de pacotes retransmitidos. Podemos observar a partir do gráfico que, embora a solução proposta atinja melhor a diferença entre eles, não foi muito alta, pois aumenta aproximadamente 3% quando comparado ao MAVEN e 10% em relação ao SFMMA.

Em resumo, a solução proposta fornece uma alta porcentagem de entrega de

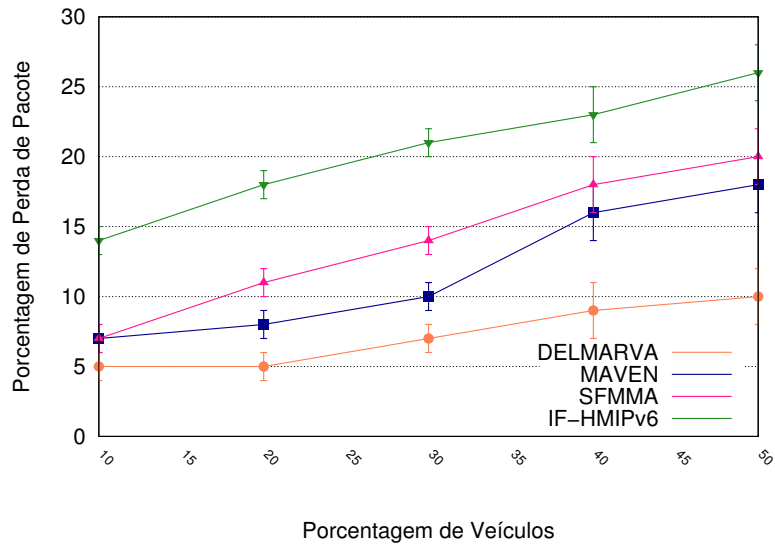


Figure 6. Porcentagem de Perda de Pacote

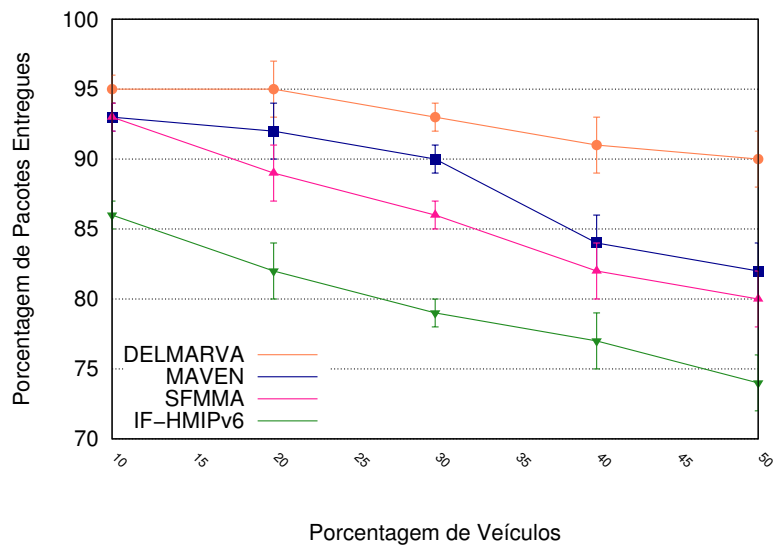


Figure 7. Porcentagem de Pacotes Entregues com Sucesso

informações sem ter que realizar muitas retransmissões, aproximadamente 90%. Além disso, o DELMARVA prove uma baixa perda de pacotes, com um baixo atraso, o que resultou em uma boa quantidade de mensagens recebidas, uma boa taxa de fluxo. Isso foi possível porque o protocolo DELMARVA executa um balanceamento de carga entre as interfaces do veículo, bem como uma intercomunicação entre as RSUs que permitiu uma diminuição no tempo de *handover*.

5. Conclusão

Neste artigo, propusemos uma nova arquitetura descentralizada para o gerenciamento de mobilidade de fluxo para as redes veiculares baseada na capacidade das RSUs para a decisão de qual é o melhor ponto de acesso que o veículo deve-se associar. A arquitetura explorou o uso de mais de uma tecnologia de rede para maximizar a QoS para aplicações em redes de veículos. Além disso, essa arquitetura trata de diferentes interfaces de rede, ao mesmo tempo, procura satisfazer os requisitos mínimos de perda de pacotes e latência para cada classe de aplicação de rede veicular, maximizando o rendimento da rede. Nós comparamos a solução proposta com outras três soluções, os resultados mostraram que nossa abordagem apresentou uma maior entrega de pacotes, aproximadamente 90%, com baixa porcentagem de perda de pacotes, bem como um baixo tempo de entrega e atraso na informação.

Como trabalhos futuros pretendemos utilizar outros parâmetros para a decisão do *handover*, como por exemplo, a energia. Além disso, desenvolveremos um mecanismo de tolerância a falhas.

Agradecimentos

Rodolfo I. Meneguette e agradece a CNPq (processos 150545/2018-5) por financiarem seus projetos de pesquisas.

References

- Balfaqih, M., Ismail, M., Nordin, R., and Balfaqih, Z. A. (2017). 802.21-assisted distributed mobility management solution in vehicular networks. *IEEE Access*, 5:9518–9532.
- Chen-wen, W. and Ping, W. (2009). Improved fast handover scheme for hierarchical mobile ipv6. In *Computer Science Education, 2009. ICCSE '09. 4th International Conference on*, pages 294–297.
- Chhabra, R., Verma, S., and Krishna, C. R. (2017). A survey on driver behavior detection techniques for intelligent transportation systems. In *2017 7th International Conference on Cloud Computing, Data Science Engineering - Confluence*, pages 36–41.
- Codeca, L., Frank, R., and Engel, T. (2015). Luxembourg sumo traffic (lust) scenario: 24 hours of mobility for vehicular networking research. In *2015 IEEE Vehicular Networking Conference (VNC)*, pages 1–8.
- Engelbrecht, A. P. (2006). *Fundamentals of computational swarm intelligence*. John Wiley & Sons.
- European Telecommunications Standards Institute (2009). Intelligent transport systems (ITS); vehicular communications; basic set of applications; definitions. Technical Report 102 638, ETSI.

- Jazar, R. N. (2017). *Vehicle dynamics: theory and application*. Springer.
- Lima, S., Barbosa, S., Palmeira, P., Matos, L., Secundo, I., and Nascimento, R. (2017). Systematic review: Techniques and methods of urban monitoring in intelligent transport systems. *ICWMC 2017*, 17:9.
- Lin, Y.-W., Hsiao, Y.-K., and Yeh, Z.-S. (2017). A new mobility management scheme for intelligent transportation systems. *Wireless Personal Communications*, 96(2):3081–3112.
- Makaya, C., Das, S., and Lin, F. (2012). Seamless data offload and flow mobility in vehicular communications networks. In *Wireless Communications and Networking Conference Workshops (WCNCW), 2012 IEEE*, pages 338–343.
- Meneguette, R. I. (2016). A vehicular cloud-based framework for the intelligent transport management of big cities. *International Journal of Distributed Sensor Networks*, 12(5):8198597.
- Meneguette, R. I., Bittencourt, L. F., and Madeira, E. R. M. (2013a). A seamless flow mobility management architecture for vehicular communication networks. *Journal of Communications and Networks*, 15(2):207–216.
- Meneguette, R. I., Boukerche, A., Guidonio, D. L., Grande, R. D., Loureiro, A. A. F., and Villas, L. A. (2016). A flow mobility management architecture based on proxy mobile ipv6 for vehicular networks. In *2016 IEEE Symposium on Computers and Communication (ISCC)*, pages 732–737.
- Meneguette, R. I., Madeira, E. R. M., and Bittencourt, L. F. (2013b). Multi-network packet scheduling based on vehicular ad hoc network applications. In *Proceedings of the 8th International Conference on Network and Service Management, CNSM '12*, pages 214–218, Laxenburg, Austria, Austria. International Federation for Information Processing.
- Ramazani, A. and Vahdat-Nejad, H. (2014). A new context-aware approach to traffic congestion estimation. In *2014 4th International Conference on Computer and Knowledge Engineering (ICCKE)*, pages 504–508.
- Sawarkar, N. D. and Madankar, M. (2016). Seamless network access by moving vehicle in heterogeneous network for vanet. *International Journal of Engineering Science*, 4170.
- Silva, F., Castillo-Lema, J., Neto, A., Silva, F., Rosa, P., Corujo, D., Guimarães, C., and Aguiar, R. (2014). Entity title architecture extensions towards advanced quality-oriented mobility control capabilities. In *2014 IEEE Symposium on Computers and Communications (ISCC)*, pages 1–6.
- Su, Z., Hui, Y., and Yang, Q. (2017). The next generation vehicular networks: A content-centric framework. *IEEE Wireless Communications*, 24(1):60–66.
- Zekri, M., Jouaber, B., and Zeghlache, D. (2012). A review on mobility management and vertical handover solutions over heterogeneous wireless networks. *Computer Communications*, 35(17):2055–2068.