

Uma Política de *Handover* de Gerência de Mobilidade de Fluxo baseada em Lógica *Fuzzy*

Rodolfo I. Meneguette¹, Luiz F. Bittencourt¹, Edmundo R. M. Madeira¹

¹Instituto de Computação - Universidade Estadual de Campinas (UNICAMP)
Av. Albert Einstein, 1251 - Cidade Universitária - Campinas/SP - Brasil

{ripolito,bit,edmundo}@ic.unicamp.br

Abstract. *Applications of vehicular ad hoc networks can take advantage of the use of simultaneous connections, thereby maximizing the throughput of the network and reducing network latency. In order to take advantage of all radio interfaces in the vehicle and to provide good quality of service for vehicular applications, we developed a handover selection policy based on fuzzy logic, which indicates the best interface for a particular flow. Each flow is the aggregation of traffic of applications in the same class. Besides indicating the best network interface of a given flow, the policy aims to minimize unnecessary changes between interfaces and data flow. We use the simulator NS3 to assess the policy performance against other approaches. For this, we compare our policy with policy-based thresholds, a policy based on game theory and a policy that also is based on fuzzy logic. It was observed that the proposed scheme showed a low flow switching time, few packet losses, and low delay.*

Resumo. *Aplicações das redes veiculares ad hoc podem tirar vantagem da utilização de conexões simultâneas, e com isso maximizam a vazão e diminuem a latência da rede. A fim de tirar proveito de todas as interfaces de rádio do veículo e para dar boa qualidade de serviço para aplicações veiculares, desenvolvemos uma política de seleção de handover baseada na lógica fuzzy, que indica qual é a melhor interface a um determinado fluxo. Cada fluxo é a agregação do tráfego de aplicações de uma mesma classe. Além de indicar a melhor interface de rede para um determinado fluxo, a política visa minimizar trocas desnecessárias entre as interfaces e os fluxos de dados. Utilizamos o simulador NS3 para analisar a política de seleção de interface. E compará-la a uma política baseada em limiares, uma política baseada na teoria dos jogos e uma política que também é baseada na lógica fuzzy. Observamos que o mecanismo proposto apresentou um tempo de troca de fluxo baixo, com menor perda de pacotes e menor atraso.*

1. Introdução

Redes ad hoc veiculares (VANET) são uma subclasse das redes móveis ad hoc que prevê a comunicação sem fio entre veículos, bem como entre veículos e dispositivos ao longo de rodovias. Em VANETs, cada veículo pode ter múltiplas interfaces de rádio, sendo assim capaz de se conectar simultaneamente a diferentes domínios e tecnologias de acesso à rede de rádio [Blanchet and Seite 2011]. Embora estes veículos possam se conectar a diferentes tecnologias de rede simultaneamente, hoje em dia os veículos estão limitados

a escolher uma interface padrão para envio e recebimento de informações. Essa limitação está relacionada com o atual modelo de gestão de múltiplas interfaces, onde várias interfaces estão ligadas ao sistema operacional [Wasserman and Seite 2011]. Normalmente, os sistemas operacionais utilizam arquivos de configuração do usuário, ou baseiam-se no tipo de aplicação para selecionar uma interface de rede padrão para enviar e receber dados [Makaya et al. 2012].

Para permitir a utilização de mais do que uma interface de rede simultaneamente, o *Internet Engineering Task Force* (IETF) tem desenvolvido a tecnologia de mobilidade de fluxo de IP, que permite a divisão de Fluxo IP entre vários enlaces de acordo com os requisitos das aplicações e preferências do usuário. Existem alguns grupos do IETF estudando as extensões de mobilidade para o IPv6 (MEXT) [Tsirtsis et al. 2011] considerando as extensões da rede de mobilidade (NETEXT) [Bernardos 2012]. Estes grupos trabalham no desenvolvimento e elaboração de um protocolo que permite o uso de mais de uma interface simultaneamente. Embora esses protocolos lidem com várias interfaces ao mesmo tempo, nenhum desses grupos de desenvolvimento especifica um protocolo ou uma política de seleção da mobilidade do fluxo, ou seja, para qual interface um determinado fluxo deverá ser mapeado e quando essa mudança deve ocorrer.

Neste trabalho propomos e avaliamos uma política de seleção utilizando lógica *fuzzy* para o mapeamento dinâmico entre as classes de aplicação das redes veiculares e as interfaces de rede ativas no veículo. A política proposta precisa lidar com as características de cada fluxo, e além disso conhecer o estado atual de cada tecnologia de rede ativa no ambiente para indicar qual a melhor interface de rede para um determinado fluxo de dados. Para isso dividimos as aplicações de redes veiculares em três classes, sendo essa divisão de acordo com os objetivos gerais de cada aplicação: *segurança*, *conforto*, e de *usuário*. Além disso, esse modelo considera que os veículos estão se movendo em uma cidade ou em uma estrada, e que os condutores ou passageiros estão executando mais de uma classe de aplicativo ao mesmo tempo. O objetivo dessa política é diminuir o tempo de troca de um fluxo de uma interface para outra, com isso diminuindo a perda de pacotes e aumentando a vazão da rede.

Este artigo está organizado como segue. Na Seção 2 descrevemos o protocolo 802.21, o protocolo *Proxy Mobile IP* versão 6 e também introduzimos conceitos de sistemas *fuzzy*. Na Seção 3 são discutidos os trabalhos relacionados. A Seção 4 apresenta a proposta de política de *handover* para a gerência de mobilidade de fluxo baseada em lógica *fuzzy*. A Seção 5 apresenta uma análise dos resultados obtidos, seguido pela conclusão na Seção 6.

2. Fundamentação Teórica

Essa seção apresenta alguns conceitos básicos utilizados neste artigo.

2.1. Protocolo 802.21

O IEEE 802.21 [Dutta et al. 2007] é um esforço recente de especificação do IEEE, que visa permitir a transferência e interoperabilidade entre redes heterogêneas, incluindo os padrões 802 e redes não 802. Uma das principais idéias do IEEE 802.21 é fornecer uma interface comum para a gestão de eventos e mensagens de controle trocadas entre dispositivos de redes que possuem tecnologias diferentes. O objetivo do IEEE 802.21 é melhorar

e facilitar o uso dos nós móveis, proporcionando transmissão ininterrupta em redes heterogêneas. Para este fim, os procedimentos de entrega podem utilizar as informações recolhidas a partir do terminal móvel e/ou infraestrutura de rede. Ao mesmo tempo, diversos fatores podem determinar a decisão de entrega: serviço de continuidade, classe de aplicações, qualidade de serviço, negociação de qualidade de serviço, segurança, etc. As tarefas mais importantes do IEEE 802.21 são: a descoberta de novas redes no ambiente e seleção da rede mais apropriada para uma determinada necessidade. A descoberta de rede e processo de seleção são facilitados pelo intercâmbio de informações da rede que ajudam os dispositivos móveis a determinar quais as redes ativas ao seu redor, permitindo assim que o dispositivo móvel se conecte à rede mais apropriada com base em suas próprias políticas [Meneguette et al. 2012]. Entretanto, esse protocolo não especifica nenhum algoritmo de seleção de rede.

O núcleo do 802.21 é a *Media Independent Handover Function* (MIHF). O MIHF terá de ser implementado em todo dispositivo compatível com o IEEE 802.21 (em *hardware* ou *software*). Essa função é responsável pela comunicação com diferentes terminais, redes e MIHFs remotos e também pelo oferecimento de serviços de informações para as camadas superiores [Marquez-Barja et al. 2011]. O MIHF define três serviços diferentes: *Independent Event Service* (MIES), *Media Independent Command Service* (MICS) e *Media Independent Information Service* (MIIS). Esses serviços permitem a obtenção e armazenamento de informações relevantes sobre o estado da rede tais como perda, vazão, e quais são as sub-redes [Meneguette and Madeira 2011]. Detalhes sobre o protocolo 802.21 podem ser encontrados em [Dutta et al. 2007].

2.2. Proxy Mobile IPv6

Proxy Mobile IPv6 (PMIPv6), conforme especificado em [Gundavelli et al. 2008], fornece um gerenciador de mobilidade baseado na rede para conectar hosts a um domínio PMIPv6. PMIPv6 introduz duas novas entidades funcionais: o *Local Mobility Anchor* (LMA) e o *Mobile Access Gateway* (MAG). MAG é a primeira camada que detecta um nó móvel (MN) associando-se a esse nó e oferecendo uma conectividade IP. O LMA é a entidade que irá atribuir um ou mais *Home Network Prefixes* (HNPs) para o nó móvel.

A base fundamental do PMIPv6 está no MIPv6 no sentido em que ele estende o MIPv6 utilizando conceitos tais como a funcionalidade do *home agent* (HA). O LMA e o MAG estabelecem um túnel bidirecional para encaminhamento de todo o tráfego de dados pertencente aos nós móveis. A gerência de mobilidade suporta uma liberdade de mobilidade dentro do domínio do PMIPv6, ou seja, um host móvel pode circular livremente dentro do domínio PMIPv6 sem alterar o seu endereço IP [Bernardos et al. 2012].

2.3. Lógica Fuzzy

Sistemas Fuzzy podem ser definidos como sistemas que utilizam a teoria de conjuntos *fuzzy* proposta por Lofti A. Zadeh em 1965 para representar pelo menos uma de suas variáveis, permitindo a representação e o processamento de informações imprecisas e incertas, abundantes no mundo real.

Os sistemas baseados em lógica *fuzzy*, ou Sistemas *Fuzzy* (SF), usam um mecanismo de raciocínio baseado no raciocínio aproximado que possui grande habilidade para expressar a ambigüidade e subjetividade presentes no raciocínio humano [Uesu et al. 2011].

As bases de regras dos SF armazenam conhecimento representado, geralmente, por meio de regras do tipo:

$$IF \rightarrow THEN$$

Os antecedentes das regras relacionam as entradas do sistema, enquanto os conseqüentes relacionam as saídas, usando operadores lógicos. Um sistema de inferência, baseado em graus de pertinência e operadores de associação, é utilizado para a obtenção de uma saída a partir de dados de entrada.

3. Trabalhos Relacionados

Esta seção apresenta alguns trabalhos relacionados a políticas de decisão no momento da troca de uma interface de rede para outra. Alguns desses trabalhos utilizam a teoria de jogos como mecanismo de decisão e outros utilizam a lógica *fuzzy*.

Patil e Kolte [Patil and Kolte 2011] desenvolveram um mecanismo de otimização de algoritmo de *handover* utilizando a lógica *fuzzy*. O algoritmo é utilizado para *handover* vertical e utiliza 3 métricas: *Signal to Interference Ratio* (SIR), a velocidade do dispositivo móvel e o tipo de tráfego da rede. Esses parâmetros são utilizados para a tomada de decisão em realizar um *handover* ou não. Essa decisão é feita pelo sistema lógico *fuzzy* que utiliza 7 tipos de saída: *Highest, Higher, High, Normal, Low, Lower, ou Lowest*.

Dhar e colaboradores [Sourav Dhar and Bera 2011] implementam um mecanismo de *handover* vertical inteligente baseado no *Analytic Hierarchy Process* (AHP) e de processos de decisão *fuzzy* para *Intelligent Transportation System* (ITS). O mecanismo utiliza o custo, a velocidade do veículo e a carga de tráfego da rede. Além disso, utiliza 2 tipos de saída na tomada de decisão do *handover*: *low* e *high*. Esse trabalho utiliza números *fuzzy* triangulares.

Dhar e colaboradores [Dhar et al. 2012] propuseram um mecanismo de *Handover*, o *Cognitive Vertical Handover* (CVHO), para assegurar a conectividade. Para isso o mecanismo utiliza rede neural artificial e o *Analytic Hierarchy Process* (AHP) para a escolha da melhor rede no momento do *handover*. Para a realização da escolha da rede a política utiliza múltiplos critérios, tais como, velocidade, largura de banda, tráfego da rede, atraso e custo.

Bin Ma e colaboradores [Ma and Liao May] propuseram um algoritmo de *handover* vertical, utilizando lógica *fuzzy* tipo 2 e levando em consideração a velocidade adaptativa do veículo na fase da descoberta de redes ativas no ambiente. Esse algoritmo adaptativo atualiza o conjunto de redes candidatas e ajusta o tempo *handover* de acordo com a velocidade do usuário. A melhor rede que faz parte desse conjunto é selecionada pela lógica Fuzzy tipo 2.

Nossa política utiliza lógica *fuzzy* para indicar a necessidade de troca de um fluxo de uma interface para outra e qual será a melhor interface. Para isso, utilizamos três variáveis lógicas: perda de pacote, vazão da rede e atraso das mensagens. Além disso, utilizamos cinco tipos de saída: muito alta, alta, normal, baixa e muito baixa, para evitar que a troca de uma interface de rede para outra seja precipitada, com isso diminuindo a quantidade de trocas executadas.

4. Política de *Handover* de Gerência de Mobilidade de Fluxo Baseada em Lógica *Fuzzy*

A política de gerenciamento de mobilidade de fluxo proposta trata do mapeamento entre o fluxo das classes de aplicações das redes veiculares e as interfaces de rede ativas no ambiente onde o veículo está transitando. Essa política tem por objetivo selecionar a melhor interface que atenda os requisitos mínimos de cada fluxo de dados. Para atingir esse objetivo utilizamos a lógica *fuzzy* como política de decisão. As regras *fuzzy* levam em conta a vazão da rede, perda de pacote e o atraso.

Essa política foi acrescentada em uma arquitetura previamente desenvolvida [Meneguette et al. 2013], que foi a base (indispensável) para desenvolvimento e avaliação desse trabalho. A Seção 4.1 apresenta uma descrição dessa arquitetura.

4.1. Arquitetura Utilizada

Essa política de seleção de fluxo foi incorporada na arquitetura de gerência de fluxo de mobilidade desenvolvida [Meneguette et al. 2012]. Essa arquitetura consiste de uma infraestrutura comum para tecnologia multi-acesso de forma transparente em redes sem fio, trabalhando com tecnologias como WiMax e LTE, além de tecnologias sem fio para redes veiculares, provendo uma conexão contínua e transparente para as aplicações veiculares.

O objetivo dessa arquitetura é maximizar a vazão da rede, mantendo a latência e a perda de pacotes dentro dos requisitos mínimos das aplicações veiculares. Para isso ela utiliza um gerenciador de fluxo baseado nas classes de aplicação das redes veiculares e no estado de cada rede ativa no ambiente.

A arquitetura utiliza o protocolo 802.21 para capturar o estado das redes ativas no ambiente e o protocolo PMIPv6 estendido. Além disso, a arquitetura está dividida em dois módulos: um módulo que está no MAG e no LMA (Figura 1(a)), e outro módulo que está embarcado no veículo (Figura 1(b)).

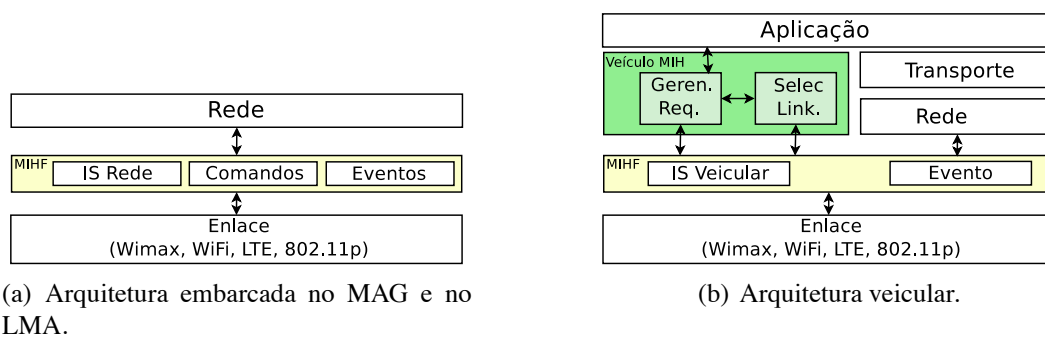


Figura 1. Componentes da arquitetura.

Na arquitetura do MAG e do LMA (Figura 1(a)), a camada de rede contém o protocolo PMIPv6 para lidar com o endereçamento e com os prefixos dos pontos de acesso requeridos para o roteamento das mensagens. Esses módulos também possuem o módulo MIHF, com funções estendidas do protocolo 802.21. Essa extensão faz com que o LMA tenha uma visão global do estado da rede que está conectada a ele, também permitindo que o MAG possa monitorar seus próprios enlaces, dando ao MAG uma visão do estado de sua rede local.

A arquitetura do veículo contém um módulo de gestão de mobilidade, chamado Veículo MIH, que compreende um módulo de gerenciamento de requisitos que recebe os requisitos mínimos de rede para que a aplicação possa ser executada. O Veículo MIH também tem um módulo de seleção de enlace, o qual recebe a informação do estado da rede e decide se vai efetuar uma transferência e, em caso afirmativo, a qual rede deve se conectar. Além disso, o módulo de seleção de enlace ajuda a interface lógica a decidir qual será o enlace usado para enviar uma mensagem específica. Essa decisão é feita através da lógica *fuzzy* proposta. Tanto o módulo de gerenciamento de requisitos quanto o módulo de seleção de enlace enviam comandos para o módulo MIHF. O módulo MIHF é uma extensão das funções do protocolo 802.21. Essa extensão permite a criação de um gerenciamento de fluxo com base nos requisitos e características de cada aplicação de redes veiculares, tais como o atraso e taxa de transferência, além de informar ao nó quais são as interfaces ativas em um determinado momento. Parte do detalhamento dessa arquitetura pode ser encontrado em [Meneguette et al. 2012].

4.2. Divisão do Fluxo de Informação

Agrupamos aplicações das redes veiculares em três classes de acordo com seus objetivos gerais: *segurança*, *conforto* e *usuário*. A classe de segurança compreende aplicações destinadas a ajudar motoristas a lidar com eventos imprevisíveis ou perigos das vias públicas através do monitoramento do tráfego próximo ao veículo através de mensagens [Singh et al. 2011]. A classe de conforto inclui aplicações que se concentram no conforto e na eficiência do fluxo de carros nas ruas e estradas. Em outras palavras, esses aplicativos aumentam o grau de conveniência de motoristas e eficiência do tráfego através da troca de informação de tráfego entre as infraestruturas de acostamento e veículos [Singh et al. 2011]. A classe de usuário é composta de aplicações que estão focadas na interação entre os ocupantes do veículo e informações, anúncios, entretenimento e vários tipos de serviço de comunicação [Meneguette et al. 2013].

Essa classificação das aplicações da rede veicular permite a divisão do fluxo de informação da rede em três fluxos distintos: o primeiro fluxo para a classe de segurança, o segundo fluxo para classe de conforto e o terceiro fluxo para a classe de usuários. Para diferenciar estes fluxos utilizamos uma 2-tupla, que consiste no protocolo usado para transmissão e a porta de destino. Em resumo, o fluxo é a agregação do tráfego das mensagens das aplicações de uma mesma classe.

4.3. Lógica *Fuzzy*

Embora o protocolo 802.21 tenha como objetivo otimizar o *handover* entre redes heterogêneas, esse protocolo não especifica um algoritmo de seleção de rede. Para resolver esse problema, desenvolvemos um mecanismo de troca de interface baseado nos requisitos mínimos de cada fluxo. Esse mecanismo utiliza lógica *fuzzy* para decidir qual interface atribuir a cada grupo e indicar se em um determinado momento algum fluxo deve mudar de interface.

Para o desenvolvimento desse mecanismo utilizamos uma técnica de número *fuzzy* triangular baseado no trabalho de Dhar e colaboradores [Sourav Dhar and Bera 2011]. A

estrutura do número *fuzzy* triangular é:

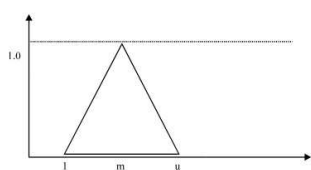
$$u_a = \begin{cases} \frac{x-l}{m-l} & l \leq x \leq m \\ \frac{u-x}{u-m} & m \leq x \leq u \\ 0 & \text{caso contrário} \end{cases}$$

sendo as variáveis l , m e u os limiares inferior, médio e superior de uma variável fuzzy. A definição acima resulta no gráfico triangular indicado na Figura 2(a), onde o eixo y representa a pertinência do valor fuzzy representado no eixo x .

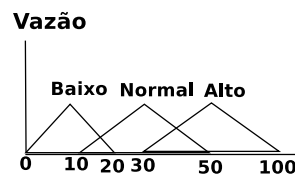
Para a criação das regras *fuzzy*, utilizamos três variáveis *fuzzy*: vazão da rede, perda de pacotes e atraso das mensagens. Cada variável possui três conjuntos *fuzzy*: alta, normal e baixa:

- Vazão: alta de 30 até 100 Kbps; normal de 10 até 50 Kbps; baixa de 0 até 20 Kbps.
- Perda: alta de 15 até 30 pacotes por segundo; normal de 5 até 20 pacotes por segundo; baixa de 0 até 10 pacotes por segundo.
- Atraso: alta de 0.3 até 1s; normal de 0.06 até 0.6s; baixa de 0 até 0.09s.

Por exemplo, os números *fuzzy* triangulares para a variável *vazão* são representados como mostra a Figura 2(b).



(a) Número triangular Fuzzy.



(b) Números triangulares fuzzy para a variável vazão.

Figura 2. Representação de números triangulares Fuzzy.

A escolha desse conjunto fuzzy e seus valores baseou-se nos padrões de avaliação de aplicações em redes veiculares da ETSI [ETSI TR 102 638 2009] e também em experimentos realizados na arquitetura ([Meneguet et al. 2012] e [Meneguet et al. 2013]).

Ao contrário do trabalho [Sourav Dhar and Bera 2011], que utiliza três classificações de saída (baixo, médio e alto), nós propomos a utilização de cinco variáveis de saída: muito alto, alto, normal, baixo e muito baixo. Optamos por um número maior de classes de saída para evitar uma precipitação no momento de realizar uma troca de interface e também fornecer um indicativo que uma interface está muito distante de atender as necessidades de um fluxo de informação. Através das combinações das variáveis de entrada com seus respectivos conjuntos, obtivemos 27 regras *fuzzy* para cada fluxo. Cada regra possui uma classificação de saída para cada tecnologia de rede. A geração dessa classificação de saída foi realizada através do fator de influência que as variáveis têm umas sobre as outras no uso de uma determinada tecnologia de rede. Para o cálculo desse fator de influência nos baseamos em [Sourav Dhar and Bera 2011]. A Tabela 1 apresenta um excerto das regras resultantes para o fluxo de segurança.

Tabela 1. Regras do fluxo de segurança

Regras	Entrada			Saída	
	Vazão da rede	Perda de pacote	Atraso	WiFi	LTE
1	Alta	Alta	Alta	Alta	Muito Baixa
2	Alta	Alta	Normal	Alta	Muito Baixa
....
9	Alta	Baixa	Baixa	Alta	Normal
10	Normal	Alta	Alta	Normal	Muito Baixa
....
18	Normal	Baixa	Baixa	Normal	Muito Alta
19	Baixa	Alta	Alta	Muito Alta	Muito Baixa
....
26	Baixa	Baixa	Normal	Alta	Muito Alta
27	Baixa	Baixa	Baixa	Alta	Muito Alta

5. Resultados

Nesta seção apresentamos as políticas utilizadas na comparação da proposta, os cenários de simulação e as análises dos resultados obtidos.

5.1. Políticas utilizadas na comparação

Utilizamos 4 políticas diferentes em nossa avaliação:

- **Proposta:** Mecanismo proposto neste trabalho (Seção 4.3).
- **Limiar:** A decisão de mudar um fluxo de uma interface para outra é realizada através de um único limiar para cada variável, ou seja, um limiar para a perda de pacote, um limiar para vazão e um limiar para o atraso dos pacotes. Quando o nó percebe que pelo menos um desses valores está passando do seu limiar, o mecanismo realiza a troca do fluxo que foi detectado com excesso para outra interface que atenda aos requisitos da aplicação.
- **Lógica Fuzzy:** A decisão de realizar a mudança de um fluxo de uma interface para outra ocorre através de inferências *fuzzy*. Para a criação desse cenário nos baseamos no trabalho de Dhar e colaboradores [Sourav Dhar and Bera 2011] utilizando somente 3 classificações de saída. Nesse cenário também utilizamos os números triangulares *fuzzy*, além das faixas das variáveis *fuzzy* de baixo, meio e alto, para melhor adequar a saída.
- **Teoria dos jogos:** Em um jogo, cada participante possui um conjunto de estratégias que podem ser usadas. Quando cada jogador escolhe qual será a sua estratégia, obtemos um perfil, ou seja, um espaço que contém todas as possíveis situações que podem ocorrer. Cada jogador tem interesses ou preferências para cada situação no jogo. Em termos matemáticos, cada jogador tem uma *função utilidade* que atribui um número real (o ganho, ou *payoff*, do jogador) a cada situação do jogo. Mais especificamente, um jogo tem os seguintes elementos básicos: um conjunto finito de jogadores, representado por $G = \{g_1, g_2, \dots, g_n\}$, onde cada jogador $g_i \in G$ possui um conjunto finito $S_i = \{s_{i1}, s_{i2}, \dots, s_{im_i}\}$ de opções, denominadas estratégias do jogador g_i ($m_i \geq 2$), onde, portanto, o jogador g_i possui m_i estratégias. Um vetor $s = (s_{1j_1}, s_{2j_2}, \dots, s_{nj_n})$, onde s_{ij_i} é uma estratégia para o

jogador $g_i \in G$, é denominado um perfil de estratégia. O conjunto de todos os perfis de estratégia pura forma o produto cartesiano:

$$\prod_{i=1}^n S_i = S_1 * S_2 * \dots * S_n,$$

denominado espaço de estratégia do jogo. Para jogadores $g_i \in G$, existe uma função utilidade

$$u_i : S \rightarrow R$$

$$s_i \rightarrow u_i(s)$$

que associa o ganho (*payoff*) $u_i(s)$ do jogador g_i a cada perfil de estratégia $s \in S$. A política de comparação que utiliza teoria de jogos é baseada no dilema do prisioneiro. Temos 3 jogadores (os 3 fluxos de dados: segurança, conforto e usuário) e temos duas estratégias, ou seja, cada estratégia seleciona uma das interfaces de rede que o usuário possui. Assumimos que um fluxo pode selecionar apenas uma interface a cada rodada. Assim temos: $G = \{g_1, g_2, g_3\}$; $S_1 = \{s_{11}; s_{12}\}$; $S_2 = \{s_{21}; s_{22}\}$; $S_3 = \{s_{31}; s_{32}\}$. A cada estratégia do jogador é atribuído um *payoff*. Para o cálculo do *payoff*, utilizamos a função de satisfação considerando os requisitos mínimos da rede, ou seja, a porcentagem que uma determinada tecnologia de rede está atendendo num determinado fluxo.

$$Payoff = \sum_{i=1}^3 w_i * \text{parâmetro}$$

O *payoff* de cada estratégia consiste na média normalizada dos parâmetros da rede tais como atraso, vazão e perda de pacotes. Os pesos w_i , $1 \leq i \leq 3$ variam no que diz respeito aos parâmetros e são calculados com base no desempenho da rede, utilizando o trabalho [Charilas et al. 2008].

5.2. Cenários de simulação e análise dos resultados

As políticas foram implementadas no Network Simulator (NS-3.13). Utilizamos o modelo PMIPv6 que foi implementado por Hyon-Young Choi [Choi et al. 2010], e também o modelo 802.21 [Salumu 2012]. A finalidade das simulações foi verificar o impacto que o nosso mecanismo de escolha causaria na rede e nas aplicações. Com isso, pretendemos verificar se o mecanismo está realizando um bom mapeamento dos fluxos de informação das classes de aplicativos para as interfaces de rede sem sobrecarregar nenhuma rede. Para isso, nós utilizamos cinco métricas para avaliar o nosso mecanismo: tempo de *handover*, perda de pacotes, atraso, atraso por classe de aplicação e vazão.

Em nosso cenário de simulação, cada veículo (ou seus ocupantes) está executando uma aplicação de cada classe, ou seja, uma aplicação da classe de segurança, uma aplicação da classe de conforto e uma aplicação da classe de usuário. A frequência de envio de mensagens de cada aplicação é baseada no padrão ETSI (European Telecommunication Standardization Institute) [ETSI TR 102 638 2009], onde a aplicação da classe de segurança envia uma mensagem a cada 0.1s, a aplicação da classe de usuário envia uma mensagem a cada 1s, e da aplicação da classe de conforto envia uma mensagem a cada 0.5s.

Conduzimos as simulações com 50 veículos que transitavam no mapa, dentre os quais selecionamos uma série de veículos para enviar e receber mensagens dos 3 tipos diferentes de aplicação. Variamos o número de veículos que estavam executando as três classes de aplicação ao mesmo tempo no conjunto {10, 20, 30, 40, 50}. Enquanto tais veículos enviavam e recebiam informações, os outros veículos apenas trafegavam nas vias sem enviar e receber mensagens.

A topologia de rede era constituída de um nó cabeado, cinco nós de backbone, um ponto de acesso LTE e três pontos de acesso 802.11p, como podemos ver na Figura 3. Todos os veículos estavam dentro da área de cobertura do LTE, porém a tecnologia WiFi não cobria todo o mapa. Para a simulação das redes 802.11p e LTE foram utilizados módulos padrões do ns3 (tal como `wifi.SetStandard(WiFi_PHY_STANDARD_80211p_SCH)` e `WifiMacHelper::Default()`). Foram realizadas 30 simulações para cada cenário para as quais calculamos intervalos de confiança de 95%.

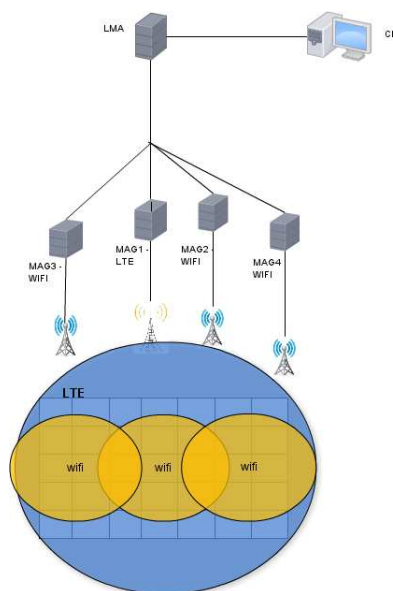


Figura 3. Topologia da simulação.

Foram utilizados dois mapas para modelar a mobilidade dos veículos. O primeiro é o modelo de mobilidade Manhattan e o segundo é um trecho de um mapa real. Para criar o modelo de mobilidade Manhattan usamos a *bonmotion*, ferramenta que criou uma grade com quatro linhas e seis colunas. Neste mapa variamos a velocidade dos veículos, entre 5, 10, 15, 20 e 25 m/s, a fim de verificar se há qualquer impacto sobre o desempenho do mecanismo proposto quando os veículos possuem diferentes velocidades. Como segundo mapa utilizamos um bairro na cidade de Campinas, no estado de São Paulo, onde o simulador de mobilidade urbana (SUMO) foi usado para converter o mapa extraído do OpenStreetMap para um formato compatível com o simulador SUMO.

Comparamos as quatro políticas descritas na Seção 5.1 (limiar, lógica fuzzy, teoria de jogos, e a nossa proposta). A Figura 4 apresenta a média dos tempos de *handover*. Analisando a Figura 4(a) podemos observar que quanto maior a velocidade do veículo maior será o tempo de *handover*. Isso ocorre devido ao aumento do número de ocorrências de *handover*. Com velocidades entre 10m/s até 25m/s todos os cenários tiveram um au-

mento significativo em seu tempo médio de *handover*. A política proposta evitou uma precipitação em trocar a interface de um determinado fluxo para outra interface, apresentando melhor desempenho. Analisando o mapa real com 30 veículos, a política *limiar* teve 32 *handovers* a mais que a política proposta. Já a política da *teoria dos jogos* teve 26 *handovers* a mais que a proposta, enquanto a política *fuzzy* fez 10 *handovers* a mais que a proposta. Comparando o mapa real com o Manhattan com 30 nós e com uma velocidade de 15 m/s a política proposta teve uma redução de 20% no tempo de *handover* comparado com o mapa Manhattan, com a proposta mantendo um desempenho melhor que as outras políticas.

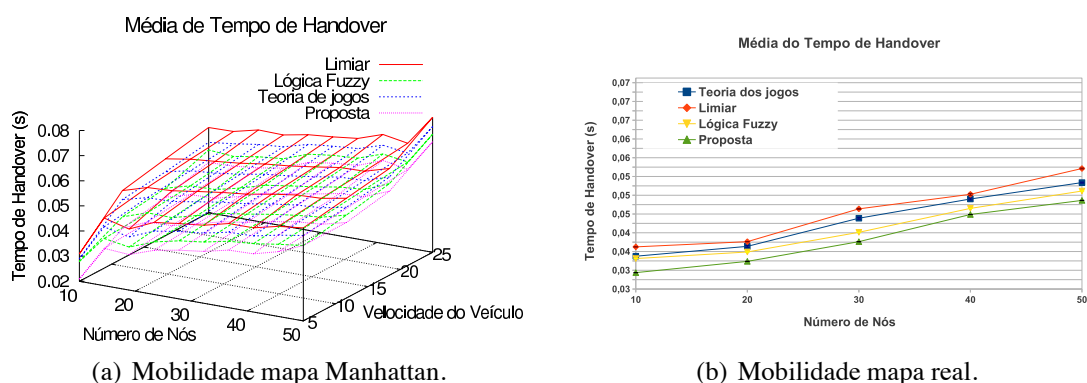


Figura 4. Média do Tempo de Handover

Podemos observar na Figura 5 que a política proposta teve uma menor perda de pacotes para ambos os mapas, devido a um baixo número de *handovers* e também por melhor distribuir os fluxos entre as interfaces. A política proposta obteve uma redução total de 18% a 38% comparada às outras políticas. Considerando 50 veículos e uma velocidade de 15 m/s, a política proposta obteve uma redução de 25% nos cenários no mapa do Manhattan e uma redução de 22% no mapa real. As taxas de perda de pacote não tiveram um impacto relevante na vazão da rede como mostra a Figura 6. Para o mapa Manhattan também não observamos impacto na vazão da rede.

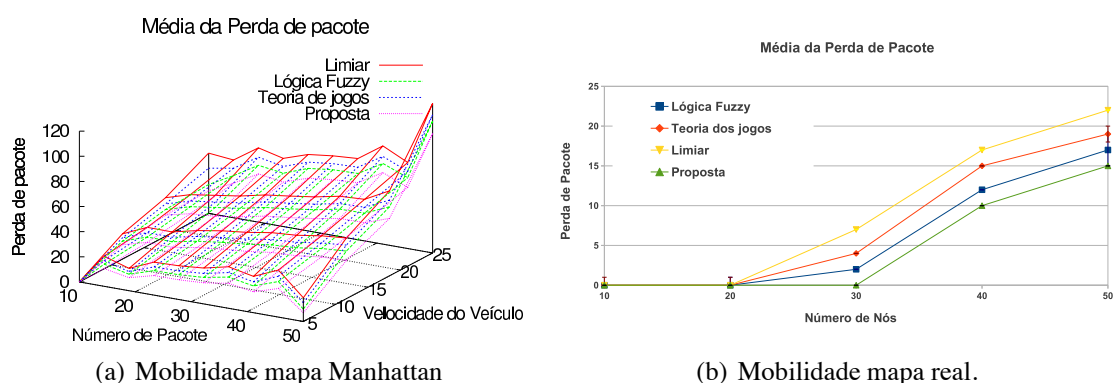


Figura 5. Média da perda de pacote

A Figura 7 apresenta a média de atraso de todas as aplicações. Na Figura 7(a) observamos que a política proposta forneceu uma redução de 6% no atraso de envio da mensagem comparado com a política *fuzzy* e uma redução de 11% comparada com a

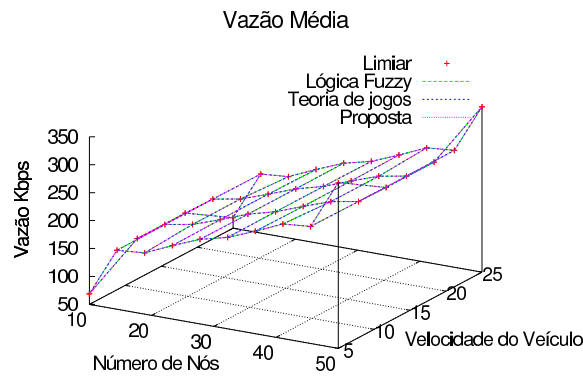
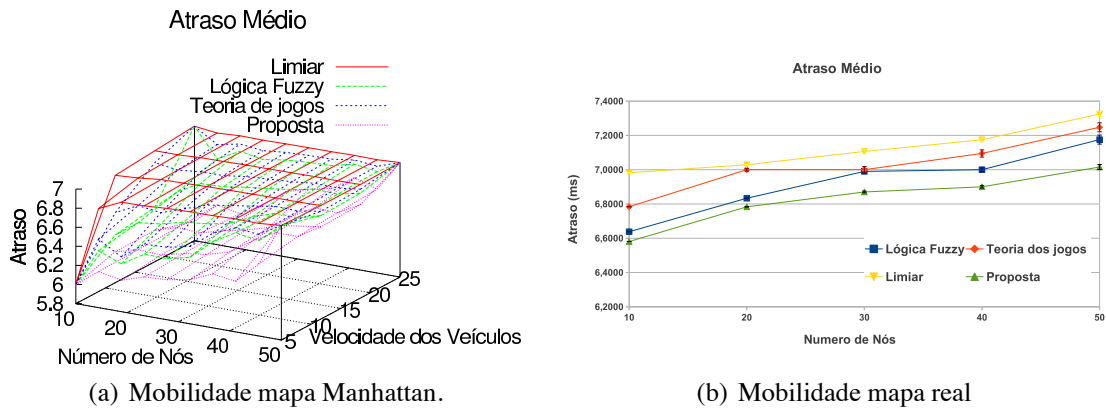


Figura 6. Vazão da rede no mapa do Manhattan em Kbps.

política *limiar* quando analisamos o gráfico com 10 veículos e com uma velocidade de 10 m/s. Na Figura 7(b), quando há 20 carros participando, a política proposta reduziu o tempo de atraso em 5% comparado à política *limiar*. Além disso, a proposta ofereceu uma redução de 67% sobre todos os cenários no gráfico 7(b). A política proposta manteve todos os atrasos por aplicação dentro de um tempo aceitável, como podemos ver na Figura 5.2. As simulações mostraram que tanto para o mapa real como para o Manhattan o atraso de todas as aplicações ficaram abaixo do padrão estabelecido pelo ETSI.



(a) Mobilidade mapa Manhattan.

(b) Mobilidade mapa real

Figura 7. Média do Atraso

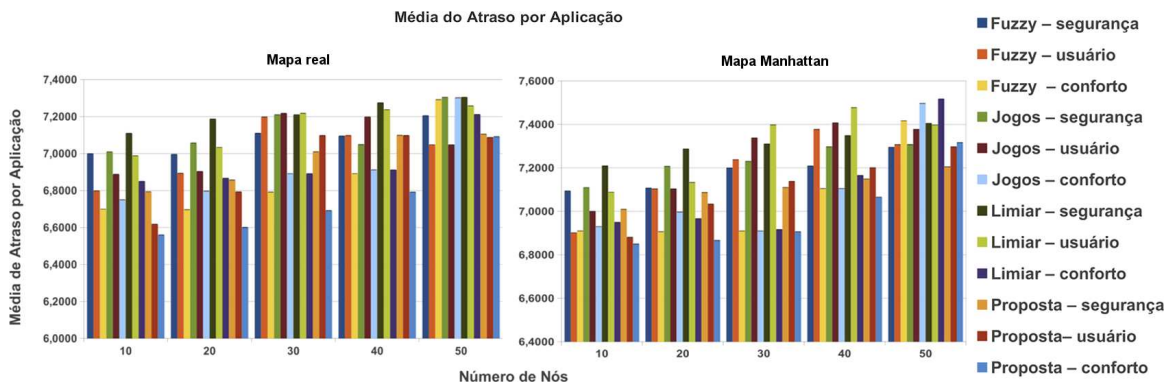


Figura 8. Média do Atraso por Aplicação.

Os resultados sugerem que a política proposta possui um bom desempenho em redes veiculares. A política proposta conseguiu uma melhoria média de 34% na perda de pacotes e de 2.8% em atraso no mapa real, enquanto para o mapa Manhattan esses números atingiram 27% e 4.95%, respectivamente. A política proposta apresentou uma baixa perda de pacote, mantendo uma boa vazão na rede e também um baixo atraso na entrega dos pacotes, não excedendo os padrões estabelecidos pelo ETSI.

6. Conclusão

Este trabalho explora políticas de decisão em redes veiculares para tratar com mudanças de fluxo de uma interface para outra num nó, considerando os requisitos de cada fluxo. A política proposta lida com o controle das interfaces de redes, buscando maximizar a vazão da rede, diminuir o tempo de troca de um fluxo de uma interface para a outra e satisfazer requisitos mínimos de perda de pacotes e atraso para cada classe de aplicação da rede veicular. Simulações mostraram que a política proposta apresentou um melhor desempenho comparado com outras políticas da literatura. O uso de mais de três classificações para a saída *fuzzy* permitiu uma melhor decisão no momento da escolha da troca do fluxo, evitando uma troca precipitada de interface e diminuindo a quantidade de *handovers* realizados na rede. Além da diminuição do tempo de *handover* a política proposta obteve uma baixa perda de pacotes com um baixo atraso quando há um grande número de participantes. Nenhum atraso das aplicações excedeu o tempo padrão estabelecido pelo ETSI.

Trabalhos futuros incluem levar consideração não somente o envio de pacote entre o veículo e a infraestrutura de acostamento, mas também entre os veículos.

Agradecimentos

Os autores agradecem CAPES, CNPq e FAPESP pelo apoio financeiro.

Referências

- Bernardos, C. J. (2012). Proxy mobile ipv6 extensions to support flow mobility. draft-ietf-netext-pmipv6-flowmob-03.
- Bernardos, C. J., Calderon, M., and Soto, I. (2012). PMIPv6 and network mobility problem statement. draft-bernardos-netext-pmipv6-nemo-ps-02.
- Blanchet, M. and Seite, P. (2011). Multiple interfaces and provisioning domains problem statement. IETF RFC 6418.
- Charilas, D., Markaki, O., Nikitopoulos, D., and Theologou, M. (2008). Packet-switched network selection with the highest qos in 4g networks. *Computer Networks*, 52(1):248 – 258.
- Choi, H.-Y., Min, S.-G., Han, Y.-H., Park, J., and Kim, H. (2010). Implementation and evaluation of proxy mobile IPv6 in NS-3 network simulator. In *5th Intl. Conference on Ubiquitous Information Technologies and Applications*, pages 1 –6.
- Dhar, S., Ray, A., and Bera, R. (2012). Cognitive vertical handover engine for vehicular communication. *Peer-to-Peer Networking and Applications*, pages 1–20.
- Dutta, A., Das, S., Famolari, D., Ohba, Y., Taniuchi, K., Fajardo, V., Lopez, R. M., Kodama, T., and Schulzrinne, H. (2007). Seamless proactive handover across heterogeneous access networks. *Wirel. Pers. Commun.*, 43:837–855.

- ETSI TR 102 638 (2009). Intelligent transport systems (its); vehicular communications; basic set of applications; definitions.
- Gundavelli, S., Leung, K., Devarapalli, V., Chowdhury, K., and Patil, B. (2008). Proxy mobile IPv6. <http://tools.ietf.org/html/rfc5213>.
- Ma, B. and Liao, X. (May). Speed-adaptive vertical handoff algorithm based on fuzzy logic in vehicular heterogeneous networks. In *9th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, pages 371–375.
- 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.
- Marquez-Barja, J., Calafate, C. T., Cano, J.-C., and Manzoni, P. (2011). An overview of vertical handover techniques: Algorithms, protocols and tools. *Computer Communications*, 34(8):985 – 997.
- Meneguette, R. and Madeira, E. (2011). An architecture for mobility management in vehicular networks with support to collaborative virtual environments. In *Brazilian Symposium on Computer Networks and Distributed Systems (SBRC 2011)*.
- Meneguette, R. I., Bittencourt, L. F., and Madeira, E. R. M. (2012). User-centric mobility management architecture for vehicular networks. In *4th International Conference on Mobile Networks and Management (MONAMI 2012)*.
- Meneguette, R. I., Bittencourt, L. F., and Madeira, E. R. M. (2013). A seamless flow mobility management architecture in vehicular communication networks. *Journal of Communications and Networks (Aceito para publicação)*.
- Patil, C. G. and Kolte, M. T. (2011). An approach for optimization of handoff algorithm using fuzzy logic system. *J. of Comp. Science and Communication*, 2(1):113–118.
- Salumu, M. (2012). ns3 - 802.21. Available at: <http://code.nsnam.org/salumu/ns-3-mih/>. Accessed in 2012.
- Singh, A., Kumar, M., Rishi, R., and Madan, D. K. (2011). A relative study of manet and vanet: Its applications, broadcasting approaches and challenging issues. In *Advances in Networks and Communications*, volume 132, pages 627–632. Springer Berlin Heidelberg.
- Sourav Dhar, Amitava Ray, S. C. and Bera, R. N. (2011). Intelligent vertical handover scheme for utopian transport scenario. *Academic Journals Inc.*
- Tsirtsis, G., Soliman, H., Montavont, N., Giaretta, G., and Kuladinithi, K. (2011). Flow bindings in mobile IPv6 and network mobility (nemo) basic support. IETF RFC 6089.
- Uesu, H., Nagashima, K., Chung, H., and Tsuda, E. (2011). Relational structure analysis of fuzzy graph and its application: For analyzing fuzzy data of human relation. In *2011 IEEE International Conference on Fuzzy Systems (FUZZ)*, pages 1593–1597.
- Wasserman, M. and Seite, P. (2011). Current practices for multiple-interface hosts. IETF RFC 6419.