

Um Novo Serviço de Gerenciamento de Tráfego para ITS baseado em Computação em Névoa

Celso A. R. L. Brennand^{1,2} e Geraldo P. R. Filho^{1,3}

¹Instituto de Computação – Universidade Estadual de Campinas

²Unidade Acadêmica de Serra Talhada – Universidade Federal Rural de Pernambuco

³Departamento de Ciência da Computação - CIC – Universidade de Brasília

celsobrennand@gmail.com e geraldoprfilho@ic.unicamp.br

Abstract. *Frustrations, monetary losses, lost time, high fuel consumption and CO₂ emissions are some of the problems caused by a traffic jam in urban centers. In an attempt to solve this problem, this article proposes a traffic service to control congestion, named uRbAN – Roteamento veiculAr na Névoa. uRbAN aims to reduce the problems generated by a traffic jam in a distributed way through suggestion of new routes to vehicles. Unlike the literature works, uRbAN is modeled using Fog computing paradigm. With this, it is possible to take advantage of the inherent aspects of this paradigm, such as low latency, load balancing, scalability, geographical location and the reduction of bandwidth usage. In order to validate uRbAN, our performance evaluation consider realistic urban scenario. When compared with others related works, uRbAN shows a reduction in stop time by up to 70 %, CO₂ emissions by up to 29 % and, planning time index by up to 49 % with satisfactory response time.*

Resumo. *Frustrações, perdas monetárias, perda de tempo, alto consumo de combustível e emissões de CO₂ são alguns dos principais problemas causados por engarrafamento nos centros urbanos. Com o intuito de sanar tais problemas, este artigo propõe o uRbAN – Roteamento veiculAr na Névoa, um serviço de tráfego para controlar os congestionamentos nos centros urbanos. O uRbAN visa reduzir os problemas gerados por um engarrafamento de forma distribuída para sugestão de novas rotas para os veículos. Ao contrário dos trabalhos da literatura, o uRbAN é modelado usando o paradigma da computação em névoa. Com isso, é possível aproveitar os aspectos inerentes de tal paradigma, tais como baixa latência, balanceamento de carga, escalabilidade, localização geográfica dos dados e a redução do uso da largura de banda. Para validar o uRbAN, nossa avaliação de desempenho foi realizada em um cenário urbano realístico. Quando comparado com outros trabalhos relacionados, o uRbAN apresenta uma redução no tempo de parada em até 70 %, na emissão de CO₂ em até 29 %, no PTI em até 49 % com um desempenho satisfatório no tempo de resposta.*

1. Introdução

O desenvolvimento não planejado dos centros urbanos está frequentemente associado aos graves problemas socioeconômicos. Tipicamente, tal crescimento causa um estresse significativo nas estruturas da cidade devido à demanda imprevista de vários recursos e

serviços. Nesse cenário, um dos setores mais afetados são os sistemas de transporte urbano, nos quais as ineficiências podem levar a muitas consequências negativas, tais como o aumento na emissão de *CO2* e um número elevado de tempo no trânsito por causa de congestionamento. No cenário internacional, o custo gerado pelo congestionamento é aproximadamente US\$ 461 bilhões [Cookson and Pishue 2017]. O Brasil está na sétima posição de país com maior índice de congestionamento, a cidade de São Paulo ocupa a quarta posição da cidade mais congestionada no mundo [Cookson and Pishue 2017].

Uma das maneiras para auxiliar na redução do problema de congestionamento, é por meio da utilização do Sistema de Transporte Inteligente (ITS). Um ITS utiliza tecnologias de comunicação, processamento e sensoriamento para melhorar o tráfego urbano e consequentemente, o fluxo de veículos nas vias urbanas. Ainda, o ITS pode ser utilizado para prover serviços de informação e entretenimento aos condutores e passageiros, de maneira a tornar o percurso deslocado um processo mais agradável [Brennand et al. 2015, Meneguette et al. 2018]. Um componente importante para a concepção do ITS é a Rede Veicular. Em tal rede, os veículos são equipados com processadores, sensores e interfaces de comunicação sem fio para comunicar entre si e com elementos pertencentes à infraestrutura da rede, formando uma rede *ad-hoc* [Meneguette et al. 2018].

Os serviços de sugestão de rotas desenvolvidos para o ITS possuem características intrínsecas em relação à maneira como processam, armazenam e disseminam dados [Rizzo et al. 2016]. Por exemplo, os dados coletados (i.e., condições de tráfego) de uma região específica podem ser irrelevantes para outras regiões da cidade. Salienta-se que para tal serviço é necessário compartilhar grandes quantidades de informações providas de fontes de dados heterogêneas como sensores no carro (e.g. GPS, velocidade) e nas vias (e.g. fluxo da via) [Rizzo et al. 2016]. Portanto, o uso de uma infraestrutura centralizada de comunicação, como a computação em nuvem [Hou et al. 2016, Ling et al. 2018], não é adequada para os serviços de sugestão de rotas devido aos atrasos na comunicação, as perdas de dados, as interrupções do canal de comunicação, além de existir problemas com a largura de banda e sobrecarga na rede.

O paradigma de computação em névoa surge como uma alternativa promissora para solucionar tais problemas. Isso ocorre em virtude da computação em névoa disponibilizar os recursos computacionais como armazenamento e processamento na borda da rede, deixando-os mais próximo do usuário final (neste caso, os serviços) sem o auxílio da Internet [Bonomi et al. 2012]. Neste trabalho, o paradigma de computação em névoa é baseado em entidades chamadas de *Cloudlet*. Tais entidades possuem capacidades de computação e comunicação (i.e., micro-data centers), além de serem geograficamente distribuídas para estarem mais próximas das redes de acesso [Satyanarayanan et al. 2009], tornando-as ideais para os requisitos exigidos para o desenvolvimento de serviços de sugestão de rotas no ITS.

Diversos trabalhos da literatura exploram o problema do gerenciamento de rotas em centros urbanos [Lin et al. 2017, Shen et al. 2018, Liebig et al. 2017, Pan et al. 2013, Jeong et al. 2016, Brennand et al. 2016, de Souza et al. 2016]. A maioria dos trabalhos realiza o monitoramento e o controle de tráfego utilizando uma arquitetura baseada na coleta de informações dos veículos e das vias. Para tanto, a arquitetura precisa trocar, processar e armazenar uma enorme quantidade de dados gerados pelos sensores incorporados nos veículos e vias. Assim, problemas relacionados ao processamento (i.e.,

balanceamento de carga e tempo de resposta) e a transmissões de dados (i.e., atrasos, perda de dados e interrupção de comunicação) tornam-se uma preocupação na comunidade acadêmica. Além disso, os serviços de sugestão de rotas possuem um tempo de resposta restrito para realizar o processo de tomada de decisão, de modo que a informação ainda seja útil para os condutores realizarem a mudança da rota.

Com base nas limitações mencionadas anteriormente, e com o intuito de avançar no estado da arte, este artigo propõe um serviço de atendimento ao tráfego para controlar o congestionamento baseado no paradigma de computação de névoa, denominado uRbAN – *Roteamento veiculAr na Névoa*. O uRbAN é baseado nas *Cloudlets* para monitorar as condições do tráfego e calcular as rotas dos veículos. Com isso, o uRbAN aproxima o poder computacional onde é mais necessário, distribuindo a carga de processamento e armazenamento do sistema. Com isso, é possível aumentar a escalabilidade geral do sistema, mantendo a capacidade de coletar, processar e armazenar grandes volumes de dados. Para isso, o uRbAN utiliza as *Road Side Units* (RSU) que são espalhadas por toda cidade como *Cloudlets* para administrar dentro de sua região o tráfego dos veículos. Para o seu funcionamento, foi desenvolvido um mecanismo que reúne todos os dados necessários. Tal mecanismo otimiza a taxa de entrega e reduz o número de mensagens. Em seguida, as *Cloudlets* usando os dados coletados, estimam o congestionamento das vias na região correspondente. Portanto, de acordo com as condições das vias que são incluídas nas rotas do veículo, as *Cloudlets* correspondentes calculam uma nova rota como sugestão. O uRbAN é comparado com outras soluções consagradas da literatura, que exploram o problema de gerenciamento de rotas em vias congestionadas. Os resultados simulados mostram que o uRbAN é mais eficiente quando comparado com tais soluções, apresentando redução no tempo de parada, na emissão de *CO2*, no PTI, além de possuir um desempenho satisfatório no tempo de resposta da infraestrutura de comunicação.

O restante deste estudo está organizado da seguinte forma. A Seção 2 apresenta os trabalhos relacionados. A Seção 3 apresenta o uRbAN e seus componentes. A Seção 4 apresenta a avaliação de desempenho do uRbAN, juntamente com a metodologia utilizada e os resultados obtidos. Finalmente, a Seção 5 apresenta as conclusões e os trabalhos futuros.

2. Trabalhos Relacionados

O problema do gerenciamento de rotas nos centros urbanos tem sido pesquisado por vários trabalhos [Lin et al. 2017, Shen et al. 2018, Liebig et al. 2017, Pan et al. 2013, Jeong et al. 2016, Brennand et al. 2016, de Souza et al. 2016] nos últimos anos, e esta seção possui o intuito de apresentar os principais desafios de pesquisa que este trabalho investiga, destacando os que não foram sanados.

Em [Pan et al. 2013], é proposto um sistema para predição, detecção e controle de congestionamentos com o intuito de reduzir o tempo de viagem, o consumo de combustível e as emissões de *CO2*. Para tanto, uma entidade centralizadora é utilizada para coletar dados em tempo real sobre a condição do tráfego de veículos. Em seguida, utiliza-se três estratégias para calcular um novo caminho: (i) *dynamic shortest path*, calcula a rota com base no menor tempo de viagem; (ii) *random k shortest paths*, seleciona uma das *k* rotas de menor caminho aleatoriamente; e (iii) *entropy balanced k shortest paths*, considera o impacto da via selecionada com base no *random k Shortest Paths* para sugerir

a nova rota. Diferente deste trabalho, as estratégias apresentadas por [Pan et al. 2013] possuem limitações, sendo as principais: (i) geração de novos congestionamentos devido a escolha da via mais livre; (ii) rotas mais longas são geradas devido à escolha aleatória; e (iii) sobrecarga de dados e comunicação devido ao uso da entidade centralizadora.

Jeong et al. [Jeong et al. 2016] propôs um sistema baseado em nuvem para otimizar o tráfego, chamado SAINT. No SAINT, os veículos relatam as condições do tráfego rodoviário a central de controle de tráfego hospedado na nuvem. O SAINT vislumbra que as RSU e eNodeB possuem acesso à Internet para comunicação com a nuvem e os veículos são equipados com 802.11p e 4G. Para redirecionar os veículos, o SAINT utiliza o algoritmo de Dijkstra modificado, no qual a função de peso leva em consideração o tempo necessário para o veículo alcançar a próxima via da rota. Assim, a probabilidade de uma rota se tornar muito popular e causar um novo congestionamento é reduzida. No entanto, o SAINT necessita informar constantemente as condições das rotas por meio da Internet para a nuvem. Além disso, a comunicação DSRC usada pelos veículos, não possui nenhum mecanismo para prover escalabilidade na comunicação, não se adequando em cenários de alta densidade como grandes cidades [De Souza et al. 2017].

Meneguette et al. [Meneguette et al. 2016] propõe o INCIDEnT, uma solução para estimar o nível de congestionamento com base em uma Rede Neural Artificial (RNA). A ideia da proposta é maximizar o fluxo de veículos nos centros urbanos. Para tanto, o INCIDEnT utiliza a velocidade do veículo e a densidade da via como entrada para a RNA estimar a condição do tráfego da via e, em seguida, sugerir novas rotas aos motoristas. Entretanto, o INCIDEnT além de não possuir o conhecimento completo do mapa, não possui um mecanismo para evitar as rotas sobrepostas que podem gerar um novo engarrafamento. Outro problema identificado está relacionado com a inexistência de um mecanismo de supressão de transmissão de dados, prejudicando assim sua eficiência, especialmente no cenário de alta densidade.

O trabalho proposto por [de Souza et al. 2016], apresenta o CHIMERA, um sistema de tráfego inteligente para maximizar a utilização da infraestrutura de transporte e reduzir o tempo médio de viagem dos veículos. Por meio de uma comunicação veicular-infraestrutura, os veículos do CHIMERA fornecem suas informações a uma RSU. Com isso, o CHIMERA realiza a detecção da via congestionada usando o algoritmo *K-Nearest Neighbor* (KNN), tendo como entrada a velocidade do veículo e a densidade da via, similar ao [Meneguette et al. 2016]. Como saída, o KNN fornece a condição do tráfego que é usada no *K-Shortest Path* para a escolha da rota. Entretanto, o CHIMERA não propôs um mecanismo de escalonamento de envio de mensagens para evitar os problemas na transmissão de dados. Além disso, a comunicação entre as RSUs no CHIMERA não é ciente das condições de tráfego de outras regiões do mapa, limitando assim a eficiência do sistema de roteamento, diferentemente do uRbAN como será apresentado a seguir.

3. uRbAN – Roteamento veiculAr na Névoa

Nesta seção é apresentado o uRbAN, um novo serviço de gerenciamento de tráfego para sugestões de novas rotas em vias que não estejam congestionadas. O uRbAN é modelado com base no paradigma da computação em névoa para distribuir os recursos computacionais (i.e., processamento, comunicação e armazenamento) entre os componentes do ITS, conforme apresentado na Figura 1. Para isso, no uRbAN, assumimos que cada RSU é

uma *Cloudlet*, e o conjunto das *Cloudlets* é que formam o ambiente computacional em névoa (Rótulo A, Figura 1). As *Cloudlets* são distribuídas no cenário de acordo com o raio de comunicação da RSU. Cada *Cloudlet* é responsável por coletar, armazenar e processar dados de uma área específica (Rótulo B, Figura 1) para computar novas rotas para os veículos em sua região. Com isso, os recursos computacionais no uRbAN são mantidos mais próximos dos usuários finais, resultando em uma melhoria significativa no sistema no que diz respeito ao processamento, comunicação e tempo de resposta. Salienta-se que as características de contexto espacial dos recursos computacionais permitem que cada *Cloudlet* seja independente uma da outra. Ainda, para aumentar a penetração de dados no ambiente e a flexibilidade na comunicação, as *Cloudlets* são equipadas com várias tecnologias de comunicação, tais como LTE e 802.11p (Rótulo C, Figura 1).

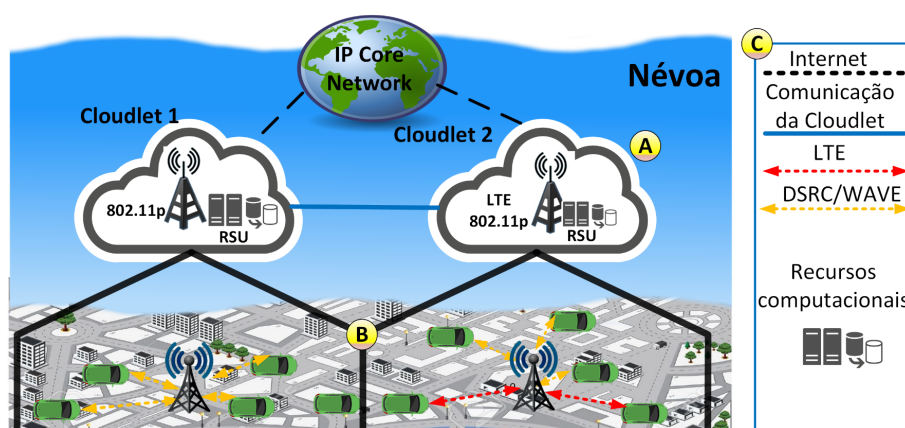


Figura 1. Cenário de funcionamento do uRbAN.

A seguir são apresentadas as três principais etapas para o desenvolvimento do uRbAN, a saber: (i) coleta de dados, Subseção 3.1; (ii) processamento de dados, Subseção 3.2; e (iii) entrega do serviço, Subseção 3.3.

3.1. Coleta de Dados

No uRbAN, um das responsabilidades das *Cloudlets* é coletar os dados dos veículos (i.e., velocidade, posição e densidade) e do ambiente urbano, como a ocupação da via. Com o intuito de obter uma cobertura global do mapa para a coleta dos dados, utilizou-se o algoritmo *Hexagonal Binning* [Carr et al. 1987] que faz uma relação entre o tamanho do cenário e o raio de comunicação das *Cloudlets*. Com isso, foi possível alcançar uma relação eficiente entre a quantidade de *Cloudlets* e o tamanho do mapa da cidade. Tal algoritmo disponibiliza os recursos computacionais mais próximo dos usuários, tornando-o uma estratégia consistente com a computação em névoa.

Para que a coleta de dados funcione adequadamente, as *Cloudlets* e os veículos, enviam *beacons* (i.e., mensagens de controle) periodicamente informando sobre as condições em sua área. A aquisição dos dados é executada de maneira distribuída usando a capacidade de comunicação das *Cloudlets*. Além disso, as *Cloudlets* enviam *beacons* informando sua posição, intervalo de roteamento para os veículos em sua área e a delimitação de sua área de responsabilidade. Um veículo usa tais informações para localizar as *Cloudlets* nas proximidades com o intuito de enviar/solicitar dados (para, por exemplo, saber para quem e quando solicitar uma nova rota). O veículo envia *beacons*

para a *Cloudlet* com dados adquiridos da sua *Onboard Unit* (OBU), informando sua rota atual, posição, velocidade e o tempo gasto para se mover em sua rota. As *Cloudlets* recebem os dados sobre uma área específica e os usam para adquirir conhecimento necessário da condição do trânsito para executar o serviço de tráfego.

Para aumentar a taxa de entrega dos *beacons*, bem como a comunicação das *Cloudlets* com os veículos, um mecanismo de agendamento de mensagens foi desenvolvido com base no padrão de comunicação DSRC/WAVE. A ideia é programar o tempo de envio das mensagens para a camada MAC 802.11p, evitando o problema de resincronização [Donato et al. 2015]. Para isso, o mecanismo agenda a hora de envio das mensagens em duas filas de acordo com o canal de comunicação que será usado para enviar a mensagem. Uma fila para o canal de serviço (SCH) que é usado para troca de mensagens padrões para o serviço (e.g. rotas dos veículos) e outra fila para o canal de controle (CCH) usado para envio de mensagens de emergência e controle do serviço. Ambas as filas são ordenadas de acordo com o tamanho da mensagem, largura de banda da interface de comunicação do veículo/*Cloudlet*, prioridade da mensagem e o intervalo de sincronização dos canais.

3.2. Processamento de Dados

Nesta etapa, o uRbAN processará as informações coletadas da etapa anterior. Como o mapa é coberto por *Cloudlets*, cada *Cloudlet* tem a responsabilidade de processar apenas dados das vias que estão dentro de sua área de cobertura. Com isso, é possível limitar o processamento de dados a uma área menor e, conseqüentemente, reduzir o custo de comunicação no uRbAN. Salienta-se, entretanto, que o tamanho da área de conhecimento que é usada para calcular a nova rota possui impacto direto na qualidade da rota. Portanto, uma área de roteamento com apenas o tamanho do raio de cobertura de rádio da RSU resultará em roteamento ineficiente. Para resolver tal limitação, a *Cloudlet* adquire informações sobre as vias que estão na cobertura de outras *Cloudlets* por meio do modelo de comunicação *request-reply*. Com isso, é possível aumentar a quantidade de informações para utilizar no algoritmo de roteamento. A área de conhecimento adicional contendo informações das vias que estão sob a responsabilidade de outras *Cloudlets* junto com as informações do mapa sob sua responsabilidade, é chamada de Área de Conhecimento (AoK) (círculo azul na Figura 2). Esta divisão é condizente com o paradigma de computação em névoa que é aplicado na modelagem do uRbAN. Em nossa solução, a AoK tem no mínimo o tamanho da área de cobertura da *Cloudlet*.

Salienta-se que a AoK é representada por um grafo ponderado e dirigido $G = (V, E)$, onde $V(G)$ são os vértices e $E(G)$ são as arestas do grafo G . Nesse caso, $V(G) = \{v_1, v_2, \dots, v_i\}$ representa o conjunto de cruzamentos dentro do alcance do AoK. Já $E(G)$ representa o conjunto de vias conectando os cruzamentos. Ainda, o peso $W = \{w_1, w_2, \dots, w_i\}$ de cada via $E_i \in E(G)$ é inversamente proporcional à velocidade com que os veículos viajam na via. Portanto, se a velocidade do veículo estiver próxima da velocidade máxima permitida na via, o W_i da via será menor. O W_i da via é usado pelo algoritmo de roteamento que será descrito na subseção 3.3. Ainda nesta etapa, a classificação da via é feita para ser usada como parâmetro de congestionamento de rota (descrito na subseção 3.3). A classificação da via é baseada nos níveis de serviços (LOS) presente no *Highway Capacity Manual (HCM)* [Board 2010]. O HCM usa a velocidade e a densidade dos veículos nas vias para estimar a capacidade e a qualidade do tráfego.

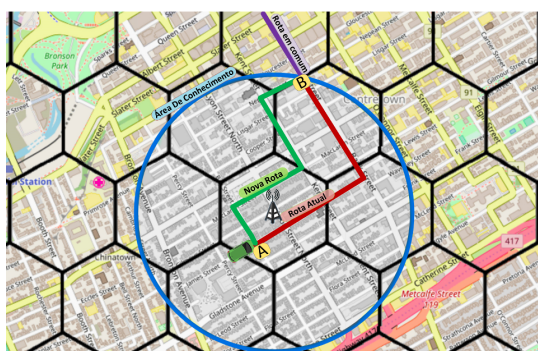


Figura 2. Área de Conhecimento no uRbAN.

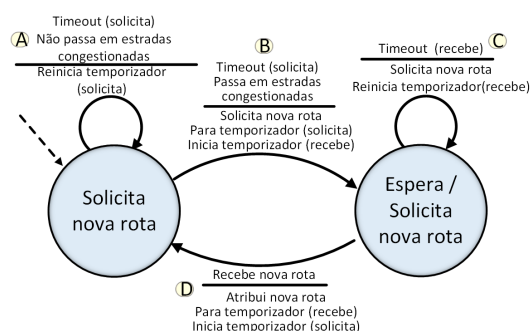


Figura 3. Diagrama de estados: Requisição de nova rota.

A qualidade do tráfego é classificada pelo HCM em 6 níveis (de A ao F), tendo o nível A como fluxo livre e degradando a qualidade até o nível F com o fluxo congestionado. O uRbAN considera uma estrada congestionada quando a velocidade da estrada é classificada como LOS C ou a ocupação da estrada é classificada como LOS D. Tais níveis de limiar foram selecionados pois é uma evidência de um congestionamento emergente. Após a classificação das vias, a *Cloudlet* dissemina por meio de *beacons* as informações das vias congestionadas que estão dentro do seu AoK, como é apresentado na próxima subseção.

3.3. Entrega do serviço

Nesta etapa, cada *Cloudlet* realiza a detecção e o controle do congestionamento calculando rotas alternativas para os veículos. A ideia é maximizar o fluxo de veículos nas vias urbanas. Cada veículo verifica periodicamente se passará por uma via de congestionamento. Para isso, os veículos recebem uma mensagem de alerta enviada pela *Cloudlet* contendo uma lista das vias congestionadas da sua AoK. Assim, a cada intervalo de roteamento, o veículo verifica se a sua rota passa por uma das vias congestionadas/[]]. Para tanto, modelou-se um mecanismo de entrega de serviço, como ilustrado na Figura 3. Se o veículo não passar por um congestionamento (Rótulo A, Figura 3), o intervalo do roteamento será reiniciado. Caso contrário, uma mensagem é enviada para a *Cloudlet* mais próxima com o intuito de solicitar uma nova rota.

Além disso, o tempo de recuperação da solicitação da rota é iniciado (Rótulo B, Figura 3). O tempo de recuperação é um mecanismo de tolerância a falhas que verifica se o veículo recebeu a rota solicitada dentro de um intervalo de tempo adequado¹. Uma nova mensagem de solicitação é enviada em caso de falha no recebimento da nova rota (Rótulo C, Figura 3). Ao receber a nova rota, o tempo do intervalo de roteamento é reiniciado (Rótulo D, Figura 3). Antes de atribuir a nova rota, a OBU do veículo solicitante verifica a satisfação da variável “fator de tamanho da rota”, que determina o quanto maior (em porcentagem) a nova rota tem em comparação com a rota atual. Desta forma, o uRbAN pode limitar o tamanho máximo da rota a ser atribuída, evitando assim o aumento do percurso e de outros problemas de tráfego como a emissão de CO₂.

Vale frisar que as *Cloudlets* calculam uma nova rota para o veículo solicitante no escopo de sua AoK, representado pelo círculo azul na Figura 2. Com isso, o processo de

¹ Definido pelo intervalo de *beacon*, pois é um tempo curto que não compromete o sistema.

re-roteamento não altera a parte da rota fora da AoK (Figura 2 – rota fixa). Como podemos ver na Figura 2, o roteamento de veículos (i.e., carro verde) é executado considerando sua posição atual (Rótulo A, Figura 2) até a última estrada em sua rota atual que está dentro da AoK da *Cloudlet* (Rótulo B, Figura 2).

O fluxograma na Figura 4 apresenta o algoritmo de cálculo da nova rota pela *Cloudlet*. A *Cloudlet* recebe uma solicitação de uma nova rota advinda de uma mensagem enviada por um veículo contendo sua posição atual e rota (Rótulo A, Figura 4). O algoritmo extrai a última via da rota que ainda está dentro da AoK da *Cloudlet* (Rótulo B, Figura 4). O restante da rota que fica fora da AoK é extraída para uso futuro (Rótulo D, Figura 4). Com isso, um conjunto de K caminhos alternativos entre a posição atual e a última via é calculado (Rótulo C, Figura 4). Uma rota deste conjunto é selecionada probabilisticamente baseada no somatório dos pesos (w) de suas vias usando a distribuição de probabilidade de Boltzmann [Kirkpatrick et al. 1983] (Rótulo E, Figura 4). Uma vez selecionada a rota, o sistema verifica se a última via da rota alternativa calculada é o destino do veículo solicitante (Rótulo F, Figura 4). Se não for, a nova rota alternativa é concatenada com o restante da rota original que fica fora da AoK da *Cloudlet* que fez o roteamento (Rótulo G, Figura 4). Entretanto, se o destino estiver na AoK, esta já será a nova rota. Com isso, uma mensagem com a nova rota é gerada e enviada para o veículo solicitante (Rótulo H, Figura 4).

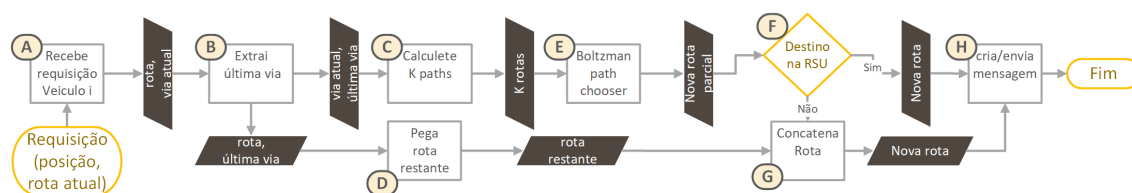


Figura 4. Fluxograma do algoritmo de roteamento do uRbAN.

4. Avaliação de Desempenho

Esta seção avalia o desempenho do uRbAN em termos da eficiência do tráfego e dos recursos computacionais da rede. Para isso, o uRbAN foi comparado com as seguintes soluções: CHIMERA [de Souza et al. 2016], PAN 1 e PAN 3 [Pan et al. 2013], *Dijkstra Shortest Path* (DSP) e com a mobilidade original do cenário, denominado *BASE*. O cenário modelado, as métricas selecionadas e a metodologia utilizada para gerar os resultados são apresentados.

4.1. Metodologia

Para avaliar o desempenho do uRbAN, foi utilizado o simulador de rede OMNeT++ 5.1². Para a simulação de tráfego e mobilidade dos veículos, utilizou-se o SUMO 0.25.0³, um simulador de tráfego aberto que permite modelar e manipular objetos no cenário urbano. Para a rede veicular, utilizou-se o *framework* Veins 4.3⁴ que implementa o IEEE 802.11p e IEEE 1609.4 DSRC/WAVE, além de um modelo de atenuação de sinal que considera obstáculos. Para medir as emissões de CO₂ e o consumo de combustível, foi

²<http://www.omnetpp.org>

³<http://sumo.dlr.de>

⁴<http://veins.car2x.org/>

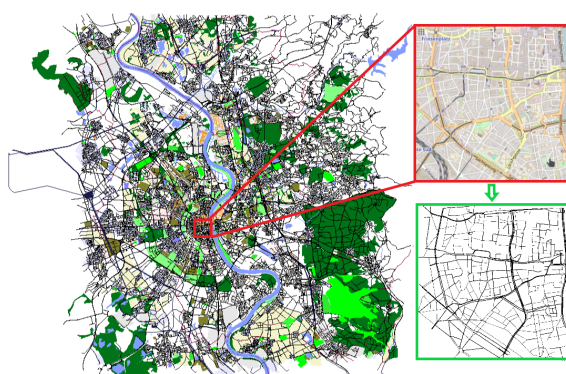


Figura 5. Topologia do cenário de Cologne.

Parâmetros	Valores
Potência de transmissão	2.2 mW
Cobertura	300 m
Taxa de transferência	18 Mbit/s
Intervalo de Beacons	4 s
Rotas alternativas (k)	3
AoK	3000 m
Intervalo de roteamento	120s

Tabela 1. Parâmetros utilizados na simulação.

utilizado o modelo EMIT⁵ do SUMO. Com o intuito de obter um cenário realista das condições de tráfego da cidade para as avaliações, foi utilizado o cenário de TAPAS Cologne [Uppoor et al. 2014] que representa a mobilidade da cidade de Colônia, Alemanha. Foi selecionada uma subárea de 4,5 km² do centro da cidade de Colônia (Figura 5) durante o período compreendido entre 7:00 e 8:00 horas. Período este que possui a maior ocorrência de congestionamento na subárea selecionada. Este cenário possui aproximadamente 14.000 veículos inseridos durante a simulação e uma extensão de 243 km de estradas.

A Tabela 1 sumariza os parâmetros utilizados na simulação. Os parâmetros da rede para todas as simulações foram ajustados para uma taxa de transferência de 18Mbit/s na camada MAC e a potência de transmissão para 2,2mW, resultando em uma cobertura de aproximadamente 300m sob um modelo de propagação de dois raios [Sommer et al. 2011]. O intervalo de roteamento foi definido em 120s para todas as soluções e são usadas três rotas alternativas para o uRbAN, CHIMERA e PAN3 com o intuito de realizar uma avaliação mais justa. Além disso, por meio de experimentos empíricos, notou-se que tal número de rotas alternativas foi o que apresentou os melhores resultados para todas as soluções nos cenários avaliados. Salienta-se que os veículos na solução DSP não divulgam informações sobre o tráfego e usam o comprimento da via como classificação da estrada. A AoK do uRbAN foi definida em 3 km.

A validação do uRbAN foi dividida em duas etapas, eficiência do tráfego e recursos computacionais da rede. Para validar a eficiência do tráfego, utilizou-se as seguintes métricas: (i) **tempo de parada**, que representa o tempo que todos os veículos ficaram preso nos engarrafamentos; (ii) **velocidade média**, que representa a velocidade média de todos os veículos; (iii) **distância percorrida**, que representa a distância média percorrida por todos os veículos; (iv) **emissão de CO2**, que representa a quantidade de CO2 emitido de todos os veículos; e (v) **Planning Time Index**, PTI, que mensura a confiabilidade do tempo de viagem obtida em relação ao fluxo ideal no mesmo caminho. Por exemplo, em uma viagem que duraria 25 minutos em tráfego livre, com o PTI no valor de 2 esta mesma viagem no mesmo trajeto durará 50 minutos. Após validar a eficiência do tráfego, validou-se os recursos computacionais da rede. Para isso, utilizou-se as seguintes métricas: (i) **total de pacotes**, que representa a quantidade de pacotes transmitidos; (ii)

⁵<http://www.hbefa.net>

colisões, que representa a porcentagem de pacotes colididos durante a simulação; (iii) **atraso da rede**, que representa o tempo médio para disseminar os pacotes para todos os veículos; (iv) **rota aceita**, que representa a quantidade de novas rotas aceitas por veículo; e (v) **rota computada por cloudlet**, que representa a quantidade de rotas calculadas por cada *Cloudlet*. Para todos os experimentos, foram executadas 33 simulações com um intervalo de confiança de 95% de acordo com o t-test. Os resultados obtidos são apresentados nas próximas subseções.

4.2. Impacto da eficiência do tráfego

A Figura 6 apresenta os resultados simulados do uRbAN, comparando-os com as soluções implementadas. Como observado na Figura 6(a), as soluções PAN1, PAN3, CHIMERA e uRbAN reduziram o tempo de parada em 62.36 %, 55.49 %, 65.62 % e 73.18 % em relação à solução BASE, respectivamente. Tais resultados refletem em uma velocidade média maior, como ratificado na Figura 6(b). Vale salientar que independente das soluções, o uRbAN apresentou um desempenho superior no que diz respeito à velocidade média, tendo um aumento de 24.92 %. Isso ocorre devido ao mecanismo probabilístico do uRbAN que seleciona a melhor rota das K rotas alternativas, além do uso da classificação da via que representa de maneira mais adequada a condição do fluxo de veículos na via. Nota-se também que o DSP obteve o tempo de parada aumentado em 63 %, e a velocidade média reduzida em 9.51 %, quando comparado com o BASE. Isto faz sentido pois, o DSP calcula apenas um caminho mais curto para todos os veículos, movendo os veículos para uma mesma estrada. Em razão disso um novo ponto de congestionamento é gerado, diferentemente da nossa solução que recalcula uma nova rota quando necessário. Entretanto, ao considerar a métrica velocidade média, o PAN1 com um aumento de 17.74 % obteve um desempenho melhor quando comparado ao PAN3 com um aumento de 13.24 % na velocidade média. Isso é coerente, posto que o PAN3 calcula três rotas alternativas e escolhe uma aleatoriamente. Dessa forma, as segundas e terceiras melhores rotas podem ter sua classificação do trânsito e tamanhos bem diferentes em comparação com a melhor rota. Salienta-se que isso não acontece no uRbAN, uma vez que a rota alternativa é selecionada através do método probabilístico implementado.

Como esperado, o DSP foi o único a reduzir a distância percorrida (2.32 %) entre as soluções avaliadas (Figura 6(c)). Observa-se que o PAN1 e o uRbAN obtiveram as menores reduções nas rotas com 1.35 % e 1.53 % em relação ao BASE, respectivamente. Já o CHIMERA obteve um aumento no tamanho da rota em 3.78 % quando comparado com o BASE, e em 2 vezes em relação ao uRbAN. Isso ocorreu porque o CHIMERA não possui um mecanismo para avaliar o tamanho da rota, diferentemente do uRbAN que implementa tal mecanismo. Na emissão de CO_2 , Figura 6(d), o uRbAN obteve uma redução de 28.25 %, o CHIMERA de 23.85 %, o PAN1 de 24.34 %, e o PAN3 de 21.12 %. Isto é, o uRbAN reduz a emissão de CO_2 independentemente das soluções comparadas. Ainda, observa-se que apesar do CHIMERA obter um tempo de parada e velocidade média melhores que o PAN1, a sua emissão de CO_2 é maior devido à distância percorrida ser maior.

A Figura 7(a) apresenta a qualidade de controle de congestionamento das soluções avaliadas utilizando a métrica PTI. Como pode ser observado na Figura 7(a), o uRbAN obteve o melhor resultado tendo o PTI reduzido em 49 % em relação ao BASE. Já o CHIMERA, o PAN1 e o PAN3 obtiveram o PTI reduzido em 44 %, em 39 % e em 35 %, respectivamente.

respectivamente. Para verificar quais soluções funcionam melhor de acordo com o tamanho da rota, a Figura 7(b) apresenta o PTI para intervalos de 500 metros. Nota-se que o uRbAN obteve um comportamento estável com o crescimento da rota, independente das soluções comparadas que obteve um aumento considerável do PTI. Portanto, tais resultados mostram que o uRbAN é capaz de lidar de maneira mais eficiente com o problema do tráfego da cidade.

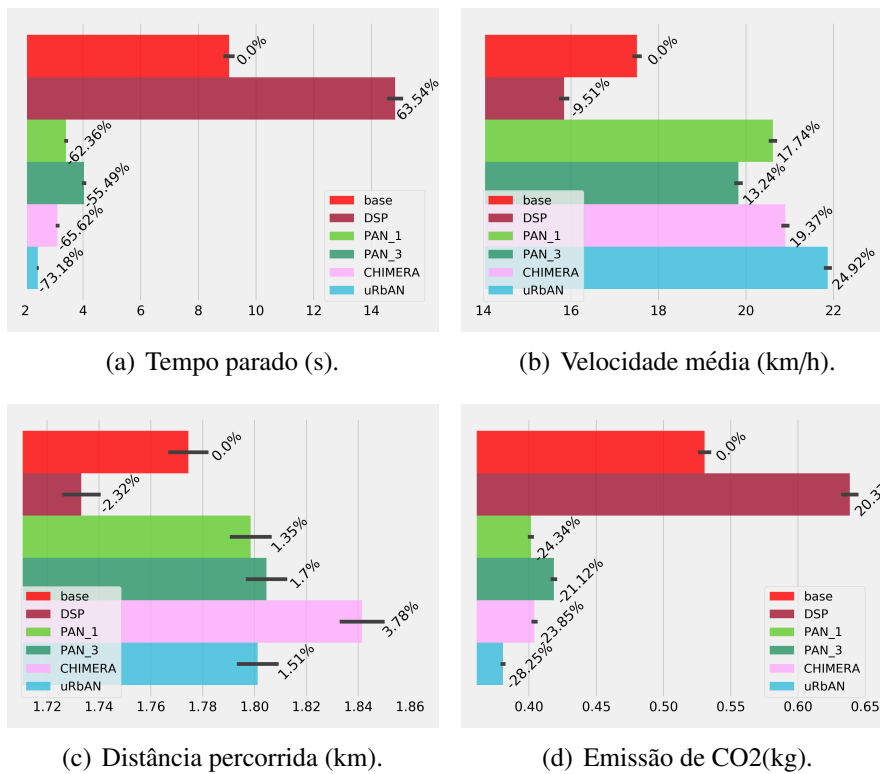


Figura 6. Eficiência do Tráfego.

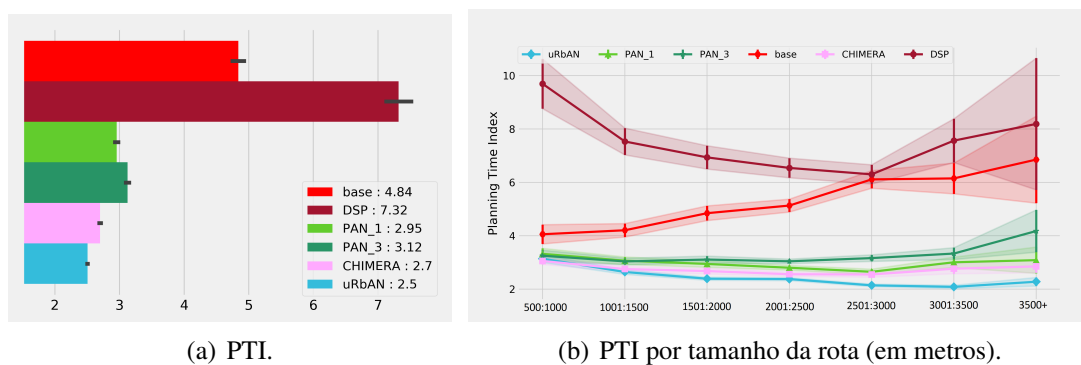


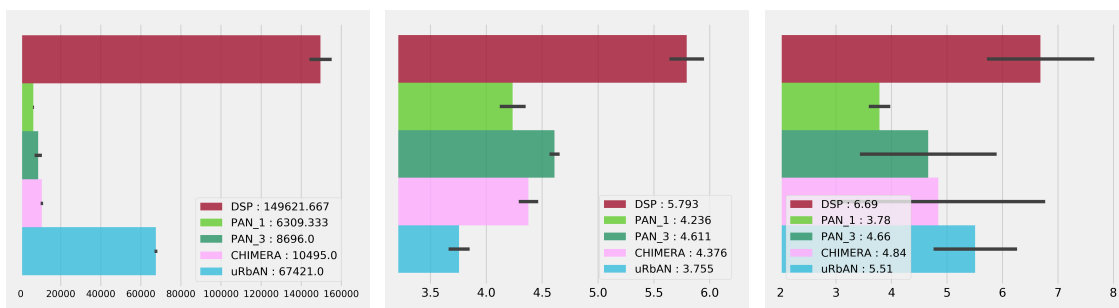
Figura 7. Impacto do PTI.

4.3. Impacto dos recursos computacionais da rede

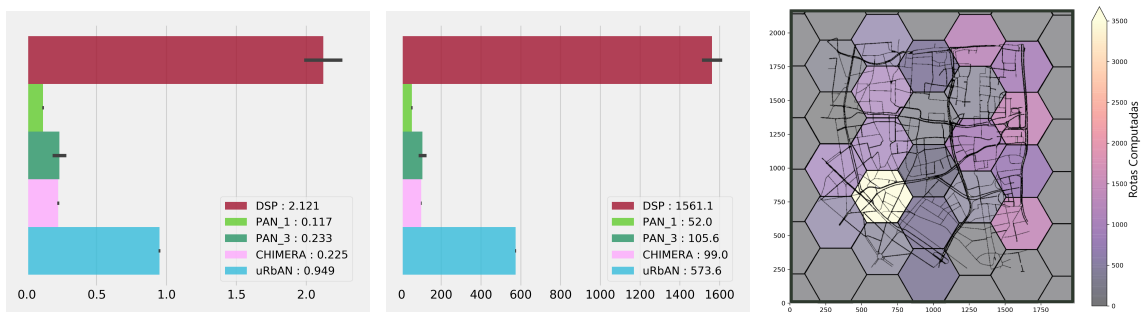
A Figura 8(a) apresenta o número total de mensagens transmitidas durante a simulação. Observa-se que o uRbAN transmite uma quantidade maior de mensagens quando comparado com o CHIMERA, PAN1 e PAN3. Entretanto, vale destacar que o uRbAN possui

menos colisões em relação ao PAN1, PAN3 e CHIMERA, Figura 8(b). Apesar do número de mensagens transmitidas pelo uRbAN ser maior, o atraso da rede para disseminar os pacotes para todos os veículos não possui diferença estatística significativa entre as soluções, como observado na Figura 8(c). Isso ocorre devido ao mecanismo de agendamento de mensagens para a camada MAC que evita o problema de ressinchronização. Com isso, é possível lidar com a concentração de veículos em uma região que concorre com o mesmo canal de comunicação da rede.

Em relação ao número de rotas aceitas pelos veículos, o uRbAN possui uma média superior, como apresentada na Figura 8(d). No pior caso, o uRbAN possui em média 0.775 vezes mais acertos por veículos re-roteados que o CHIMERA. Já no melhor caso, o uRbAN possui 0.829 vezes mais acertos por veículos re-roteados que o PAN1. De maneira geral, um maior número de veículos roteados de maneira correta, possui um aumento na qualidade do tráfego da cidade. A Figura 8(e) apresenta as médias das rotas



(a) Total de pacotes enviados. (b) Colisão por pacote enviado(%). (c) Atraso de rede(ms).



(d) Média de rotas aceitas por veículo. (e) Média de rotas computadas por RSU. (f) Computação de rotas por região.

Figura 8. Custo de Rede e Recursos Computacionais.

computadas por cada *cloudlet*. O uRbAN obteve 573.6 rotas computadas, enquanto o PAN1, PAN3 e CHIMERA obtiveram 52, 105.6 e 99 rotas computadas, respectivamente. O elevado número de rotas computadas não prejudica o desempenho do uRbAN por causa do uso do paradigma de computação em névoa, o qual distribui o cálculo das rotas por regiões em várias *Cloudlets*. Isso pode ser confirmado pela Figura 8(f), que apresenta geograficamente a distribuição dos cálculos das rotas do uRbAN no mapa, expondo as áreas com maior demanda por *Cloudlets*. Em outras palavras, o uRbAN distribui a carga de processamento e armazenamento do sistema onde é mais necessário. Portanto, tais resultados mostram que o uRbAN é mais adequado para lidar com congestionamentos de tráfego nos cenários avaliados.

5. Conclusão

Este artigo propôs o uRbAN, um serviço de atendimento ao tráfego para controlar o congestionamento baseado no paradigma de computação de névoa. O uso do paradigma de computação em névoa apresentou diversos benefícios para o uRbAN, como distribuição computacional, fornecendo o balanceamento de carga de processamento e a proximidade dos recursos computacionais com os usuários, diminuindo o tempo de resposta e o uso da largura de banda da rede. Os resultados simulados mostram que o uRbAN é mais eficiente em diversos cenários quando comparado com outras soluções da literatura. Os resultados mostraram ser relevantes pois apresentaram melhorias significativas no tráfego urbano com redução no tempo de parada, emissão de CO_2 e PTI. Em relação ao custo computacional da rede, o uRbAN também mostrou melhorias significativas reduzindo a colisão de pacotes com um menor tempo de resposta. Como trabalho futuro, pretende-se realizar experimentos considerando outros cenários e o uso de outras tecnologias de comunicação como LTE.

Referências

- Board, T. R. (2010). *HCM 2010 - Highway capacity manual*. National Research Council.
- Bonomi, F., Milito, R., Zhu, J., and Addepalli, S. (2012). Fog computing and its role in the internet of things. In *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*, pages 13–16. ACM.
- Brennand, C. A., Cunha, F., Maia, G., Cerqueira, E., Loureiro, A. A., and Villas, L. A. (2016). Fox: A traffic management system of computer-based vehicles fog. In *2016 IEEE Symposium on Computers and Communication (ISCC)*, pages 1–6. IEEE.
- Brennand, C. A., de Souza, A. M., Maia, G., Boukerche, A., Ramos, H., Loureiro, A. A., and Villas, L. A. (2015). An intelligent transportation system for detection and control of congested roads in urban centers. In *2015 IEEE Symposium on Computers and Communication (ISCC '15)*, pages 663–668. IEEE.
- Carr, D. B., Littlefield, R. J., Nicholson, W. L., and Littlefield, J. S. (1987). Scatterplot matrix techniques for large n. *Journal of the American Statistical Association*, 82(398):424–436.
- Cookson, G. and Pishue, B. (2017). Inrix global traffic scorecard.
- De Souza, A. M., Brennand, C. A., Yokoyama, R. S., Donato, E. A., Madeira, E. R., and Villas, L. A. (2017). Traffic management systems: A classification, review, challenges, and future perspectives. *International Journal of Distributed Sensor Networks*, 13(4):1550147716683612.
- de Souza, A. M., Yokoyama, R. S., Maia, G., Loureiro, A., and Villas, L. (2016). Real-time path planning to prevent traffic jam through an intelligent transportation system. In *Computers and Communication (ISCC), 2016 IEEE Symposium on*, pages 726–731. IEEE.
- Donato, E., Maia, G., Duarte, J. M., Loureiro, A. A., Madeira, E., and Villas, L. (2015). Presync: A method for preventing resynchronization in the IEEE 802.11 p standard. In *Computers and Communication (ISCC), 2015 IEEE Symposium on*, pages 457–462. IEEE.

- Hou, X., Li, Y., Chen, M., Wu, D., Jin, D., and Chen, S. (2016). Vehicular fog computing: A viewpoint of vehicles as the infrastructures. *IEEE Transactions on Vehicular Technology*, 65(6):3860–3873.
- Jeong, J., Jeong, H., Lee, E., Oh, T., and Du, D. H. (2016). Saint: Self-adaptive interactive navigation tool for cloud-based vehicular traffic optimization. *IEEE Transactions on Vehicular Technology*, 65(6):4053–4067.
- Kirkpatrick, S., Gelatt, C. D., Vecchi, M. P., et al. (1983). Optimization by simulated annealing. *science*, 220(4598):671–680.
- Liebig, T., Piatkowski, N., Bockermann, C., and Morik, K. (2017). Dynamic route planning with real-time traffic predictions. *Information Systems*, 64:258–265.
- Lin, J., Yu, W., Yang, X., Yang, Q., Fu, X., and Zhao, W. (2017). A real-time en-route route guidance decision scheme for transportation-based cyberphysical systems. *IEEE Transactions on Vehicular Technology*, 66(3):2551–2566.
- Ling, C. W., Datta, A., and Xu, J. (2018). A case for distributed multilevel storage infrastructure for visual surveillance in intelligent transportation networks. *IEEE Internet Computing*, 22(1):42–51.
- Meneguette, R. I., Filho, G. P. R., Guidoni, D. L., Pessin, G., Villas, L. A., and Ueyama, J. (2016). Increasing intelligence in inter-vehicle communications to reduce traffic congestions: Experiments in urban and highway environments. *PLOS ONE*, 11(8):1–25.
- Meneguette, R. I., Robson, E., and Loureiro, A. A. F. (2018). *Intelligent Transport System in Smart Cities: Aspects and Challenges of Vehicular Networks and Cloud*. Springer.
- Pan, J., Popa, I. S., Zeitouni, K., and Borcea, C. (2013). Proactive vehicular traffic rerouting for lower travel time. *IEEE Transactions on vehicular technology*, 62(8):3551–3568.
- Rizzo, G., Palattella, M. R., Braun, T., and Engel, T. (2016). Content and context aware strategies for qos support in vanets. In *2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA)*, pages 717–723. IEEE.
- Satyanarayanan, M., Bahl, P., Caceres, R., and Davies, N. (2009). The case for vm-based cloudlets in mobile computing. *IEEE pervasive Computing*, 8(4).
- Shen, Y., Lee, J., Jeong, H., Jeong, J., Lee, E., and Du, D. H. (2018). Saint+: Self-adaptive interactive navigation tool+ for emergency service delivery optimization. *IEEE Transactions on Intelligent Transportation Systems*, 19(4):1038–1053.
- Sommer, C., German, R., and Dressler, F. (2011). Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis. *IEEE Transactions on Mobile Computing*, 10(1):3–15.
- Uppoor, S., Trullols-Cruces, O., Fiore, M., and Barcelo-Ordinas, J. M. (2014). Generation and analysis of a large-scale urban vehicular mobility dataset. *IEEE Transactions on Mobile Computing*, 13(5):1061–1075.