

# DINO: Roteamento Dinâmico para ITS Considerando Recursos da Rede

Matheus Bernardino de Araújo  
Universidade Estadual do Ceará  
Fortaleza, Brasil  
matheus.bernardino@larces.uece.br

Matheus Monteiro Silveira  
Universidade Estadual do Ceará  
Fortaleza, Brasil  
matheus.monteiro@larces.uece.br

Rafael Lopes Gomes  
Universidade Estadual do Ceará  
Fortaleza, Brasil  
rafaellgom@larces.uece.br

## ABSTRACT

Intelligent Transport Systems (ITS) arose as a modern solution to traffic jams and vehicle accidents in the urban environment. A key part of the ITS is Traffic Management (TM), which concerns the planning and route definition of the vehicle. Existing TM solution focuses specifically on urban traffic information, ignoring the issues related to the network infrastructure and the applications at the top of it. Within this context, this paper presents a vehicle routing and re-routing strategy, called DINO, that considers both travel time of vehicles on the roads and the active network flows in the network, aiming to dynamically bring a suitable balance between travel time and packet delivery through a heuristic. The experiments performed suggest that DINO improves the packet delivery of the applications while reduces the average travel time of vehicles.

## KEYWORDS

Redes Veiculares, Sistemas de Transporte Inteligente, LTE, Problema de Roteamento de Veículos.

## 1 INTRODUÇÃO

O transporte urbano tem se tornado nos últimos anos um tema do cotidiano de quem mora nas cidades de grande e médio porte. Os congestionamentos, os acidentes de trânsito, a baixa qualidade do transporte público e as dificuldades de estacionamento são somente alguns dos transtornos enfrentados nos ambientes urbanos modernos. Mesmo dispondo-se de recursos, é inviável a expansão das vias na mesma proporção, visto os efeitos ambientais e energéticos que essa expansão provocaria.

Uma possível solução para esses problemas é pela redução significativa dos deslocamentos realizados por automóveis e sua respectiva transferência para o transporte coletivo, que é um processo de longa maturação que envolve a integração das políticas públicas de transporte, desenvolvimento urbano e políticas econômicas. Além disso, essa solução necessita de grandes investimentos na melhoria da qualidade do transporte coletivo, capazes de atrair os usuários do transporte individual. Por outro lado, a aplicação da tecnologia da informação, aliada ao planejamento do transporte urbano, tem se configurado como uma alternativa viável em termos de custo-benefício que possa resolver esses problemas. Esta nova abordagem, chamada de Sistemas de Transporte Inteligentes (*Intelligent Transportation Systems - ITS*) [1], encontra-se em expansão nos países desenvolvidos.

Uma parte importante do ITS é o Gerenciador de Tráfego de Veículos (*Traffic Manager - TM*), que diz respeito ao planejamento e definição da rota dos veículos no meio urbano. A definição da rota pode ser orientada por vários critérios, como tempo de viagem, congestionamento urbano, emissão de dióxido de carbono ( $CO_2$ ) e

outros. Para realizar uma TM eficiente, é necessário coordenar os veículos de forma colaborativa, bem como apoiar a comunicação entre a TM do ITS e os veículos. Uma das tecnologias que surgiram como uma opção eficiente de suporte à TM são as Redes ad-hoc Veiculares (*Vehicle ad-hoc Networks - VANETs*).

Uma abordagem popular para possibilitar a coordenação em ambientes urbanos é o IEEE 802.11p [2], ao prover a comunicação entre os veículos para aplicações ITS em VANETs. Entre suas vantagens estão fácil implantação, baixo custo, tecnologia madura e capacidade de suportar nativamente comunicações Veículo-a-Veículo (*Vehicle-to-Vehicle - V2V*) em modo ad-hoc. No entanto, esta tecnologia sofre de problemas de escalabilidade, atrasos imprevisíveis e falta de garantias determinísticas de qualidade de serviço (*Quality of Service - QoS*) [3]. Além disso, devido ao seu alcance limitado de rádio, o 802.11p só pode oferecer conectividade entre veículos e a infraestrutura (*Vehicle-to-Infrastructure - V2I*) de forma intermitente e de curta duração.

A partir das desvantagens mencionadas, o *Long Term Evolution - (LTE)* surgiu como uma potencial tecnologia sem fio de acesso para apoiar a comunicação entre veículos e ITS. O LTE é uma tecnologia de banda larga sem fio que oferece alta taxa de dados e baixa latência para usuários móveis. Pode se beneficiar de uma grande área de cobertura, alta taxa de penetração e suporte de terminal de alta velocidade, tornando sua utilização viável para o contexto de ITS [4].

Contudo, o uso de LTE em ambientes ITS e VANET não é trivial. Para fornecer serviços confiáveis aos veículos, todos os atores em Veículo-para-Tudo (*Vehicle-to-Everything - V2X*) devem frequentemente transmitir pequenas informações de algumas centenas de bytes, geralmente a cada 0, 1 a 1 segundo [5]. Essas características de tráfego são diferentes das comunicações celulares típicas, onde um número relativamente pequeno de usuários está ativo ao mesmo tempo [6].

Adicionalmente, ao longo dos anos, o número de dispositivos conectados às redes LTE aumentou, devido ao acréscimo da alta densidade de veículos em vias congestionadas e aos pedestres concentrados nas ruas ou dentro dos veículos [7]. Este massivo número de conexões e seu volume de tráfego em torres LTE, tendem a esgotar os recursos da rede, comprometendo a QoS do ITS.

Para evitar congestionamentos das torres *Evolved Node B - (eNB)* da rede LTE, que são responsáveis pela comunicação direta via rádio com os veículos, este trabalho apresenta uma estratégia de roteamento e re-roteamento de veículos, denominada DINO, que considera tanto o tempo de viagem dos veículos nas vias quanto o fluxo de rede ativo nas torres LTE para gerar as rotas. O objetivo do DINO é definir rotas e fazer o re-roteamento de veículos de forma dinâmica que proporcionem um equilíbrio adequado entre o tempo

de viagem e a entrega dos pacotes das aplicações, além de reduzir a sobrecarga nas torres eNB.

Pelos experimentos realizados, foi possível perceber que as rotas definidas pelo DINO melhoram, quando comparadas às abordagens existentes, a entrega de pacotes das aplicações ao mesmo tempo em que reduzem o tempo médio de viagem dos veículos.

O restante deste artigo está organizado da seguinte forma. A seção 2 detalha os trabalhos existentes relacionados à mobilidade urbana, ITS e estratégias de TM. A seção 3 descreve a solução DINO e seus objetivos. A seção 4 apresenta as configurações do cenário utilizado nos experimentos, enquanto a seção 5 mostra os resultados dos experimentos realizados. Finalmente, a seção 6 conclui o artigo.

## 2 TRABALHOS RELACIONADOS

Nesta seção, descrevemos os principais trabalhos relacionados sobre mobilidade urbana, ITS e estratégias de TM. Várias soluções existentes focam exclusivamente em informações de tráfego urbano, como as referências [7–11]. Por outro lado, existem soluções que incorporam parâmetros de rede à estratégia de TM, por exemplo, as propostas apresentadas nas referências [5, 12]. A Tabela 1 resume esses trabalhos existentes na literatura, destacando as diferenças em relação à nossa proposta.

Akanabe et al. [8] propõem uma abordagem, chamada APOLO, que emprega conhecimento histórico dos padrões de mobilidade dos motoristas para obter uma visão global da rede viária. A ferramenta APOLO não considera a troca constante de informações entre os veículos e a existência de um servidor central para obter uma visão global do estado do tráfego rodoviário. Porém, o APOLO depende diretamente das informações históricas da população do ambiente urbano, limitando sua aplicabilidade em ambientes urbanos sem esses dados e com variação da mobilidade.

Pessoa et al. [9] descrevem um *framework* para resolver problemas de roteamento de veículos (VRPs) com base em algoritmos de corte (*Branch-Cut-Price* - BCP). A proposta incorpora os seguintes elementos: relaxamento de caminhos, cortes de classificação com memória limitada, enumeração de caminho e cortes de capacidade arredondada. A partir desses elementos, os autores generalizaram características dos ambientes urbanos, a fim de derivar suposições sobre congestionamento de vias e possíveis soluções. Essas suposições comprometem as abordagens de implantação com reconhecimento de rede, bem como para realizar o redirecionamento de veículos. Além disso, os autores não consideram os requisitos dos serviços e aplicações dos veículos, tampouco a capacidade da infraestrutura de rede que dá suporte a estes serviços e aplicações.

Ruiz et al. [10] projetaram um algoritmo genético de chave aleatória para resolver o problema de roteamento de veículos com restrições. A função objetivo do problema é minimizar a distância total percorrida pelos veículos, respeitando as restrições de capacidade e distância máxima a ser percorrida. Assim, o algoritmo genético projetado não considera as questões de comunicação, nem a disponibilidade dos recursos da rede para as aplicações.

Rubin et al. [5] desenvolveram dois mecanismos de gestão de tráfego e redes de dados, integrando-os em um sistema de transporte autônomo. No primeiro mecanismo, os veículos que se deslocam em cada pista são organizados em pelotões, onde um Líder de Pelotão (PL) é eleito para gerenciar seus membros e suas comunicações

com a infraestrutura e com os veículos de outros pelotões. Posteriormente, no segundo mecanismo, certos líderes de pelotão são atribuídos dinamicamente para atuar como nós de *backbone* (BNs) para formar uma rede de backbone (Bnet). Os mecanismos propostos visam implantar uma infraestrutura de rede sem fio de base, mas os autores não consideram a densidade e a disponibilidade dos recursos da rede para realizar o roteamento e re-roteamento dos veículos.

Guidoni et al. [7] apresentam uma solução de TM com base nas informações sobre a densidade de veículos nas estradas e o modelo macroscópico de engenharia de tráfego densidade-fluxo. Essas informações auxiliam na modelagem da identificação de rotas congestionadas e, conseqüentemente, na redução do tempo de viagem dos veículos. A solução tem como foco a redução de congestionamentos, desconsiderando a demanda de rede das aplicações em execução no ambiente urbano.

Souza et al. [11] apresentam CHIMERA, um sistema de transporte inteligente para melhorar a utilização espacial geral de uma rede rodoviária e reduzir os custos médios de viagem de veículos. O principal foco deste trabalho é evitar que os veículos fiquem presos no trânsito e maximizar a velocidade média de deslocamento dos mesmos. Apesar de realizar o roteamento e o re-roteamento de veículos, este trabalho utiliza como critério apenas as condições do tráfego rodoviário, sendo incapaz de trazer o equilíbrio entre congestionamento de tráfego e disponibilidade de recursos de rede esperado para ITS modernos.

A partir do levantamento bibliográfico realizado, nenhum dos trabalhos encontrados foca no problema de roteamento e re-roteamento de veículos com base tanto no congestionamento das vias quanto na disponibilidade de recursos de rede em torres LTE. Assim, este artigo apresenta uma solução para trazer um equilíbrio adequado entre o tempo de viagem e a entrega de pacotes das aplicações no topo do ambiente urbano.

## 3 PROPOSTA

Os dados de fluxo de tráfego e o tempo de viagem nas vias podem ser determinados em tempo real com base nos dados relatados pelos veículos [13]. Esse tipo de informação pode ser enviada em forma de **Cooperative Awareness Messages** (CAMs) [14]: mensagens periódicas transmitidas para fornecer o status do veículo com uma frequência de 10 Hz, latência máxima de 100 ms e tamanho de até cerca de 800 bytes.

Um servidor ITS central recebe tais informações a partir dos veículos e fornece controle centralizado para outras entidades, além de informações sobre tráfego, vias e serviços. Normalmente, o servidor ITS pode ser implantado conectado à rede LTE (como uma abordagem de Névoa) ou na Nuvem (que será acessada através da rede LTE) [5]. Nossa proposta independe da localização do servidor ITS.

De acordo com os experimentos apresentados em [15], as redes LTE são incapazes de satisfazer os requisitos de entrega de mensagens do tipo CAM quando o eNB tem recursos de rede limitados, uma vez que as redes LTE são a opção de transmissão de dados de muitas aplicações, como conteúdo multimídia, chamadas de voz, jogos para celular online, mensagens de texto, sistemas de segurança e outros [12].

Table 1: Trabalhos Relacionados

Referência	Recursos de Rede	Rotas Dinâmicas	Proposta
Akanabe et al. [8]	Não	Não	Utilização do conhecimento histórico dos padrões de mobilidade urbana das estradas.
Pessoa et al. [9]	Não	Não	Modelagem genérica de um solver para VRPs com base em algoritmos BCP.
Ruiz et al. [10]	Não	Não	Algoritmo genético para resolver o VRP com restrições de capacidade e distância.
Rubin et al. [5]	Não	Não	Eleição de uma PLs que fará parte de uma Bnet para apoiar à comunicação.
Guidoni et al. [7]	Não	Sim	TM baseado na densidade de veículos nas estradas e sua densidade de fluxo.
Souza et al. [11]	Não	Sim	Redução de custos de viagem e congestionamento por meio da utilização espacial de estradas.
Este trabalho	Sim	Sim	TM com base no congestionamento de estradas e na disponibilidade de recursos de rede.

Nesse contexto, é necessário rotear (e re-roteamento) os veículos considerando o congestionamento das vias urbanas e a disponibilidade de recursos de rede nas torres LTE, que é o objetivo da estratégia DINO, nossa proposta de TM para ITS. A seguir, descrevemos o modelo de comunicação, a estratégia de roteamento de veículos e a abordagem de balanceamento de tráfego que são aplicadas pelo DINO nas Seções 3.1, 3.2 e 3.3, respectivamente.

### 3.1 Modelação do Problema

A rede de vias é representada pelo grafo direcionado ponderado  $G = (V, E)$ , em que o conjunto de vértices  $V = \{v_1, v_2, \dots, v_n\}$  corresponde às interseções entre as vias, sendo estas representadas pelo conjunto de arestas  $E = \{e_1, e_2, \dots, e_m\}$ . Além disso,  $W = \{w_1, w_2, \dots, w_m\}$  é o conjunto de pesos de cada aresta em  $E$ . O conjunto  $\mathfrak{V} = \{v_1, v_2, \dots, v_k\}$  é o conjunto de veículos e  $R = \{r_1, r_2, \dots, r_k\}$  é o conjunto de rotas para cada veículo.

Por fim, o conjunto  $T = \{t_1, t_2, \dots, t_z\}$  representa as torres eNB utilizadas para o processo de comunicação dos veículos em  $\mathfrak{V}$ . Cada torre de  $T$  individualmente cobre uma área específica e o conjunto com todas as torres cobre todo o cenário. Dessa forma, cada torre  $t_i$  possui, associado a ela, um subconjunto de vias que estão cobertas pelo sinal desta torre. As vias posicionadas na interseção de duas ou mais torres ficam associadas a que estiver menos congestionada, ou seja, que estiver transmitindo menos bytes. As vias posicionadas na interseção de duas ou mais torres estão associadas com a torre que possuir mais recursos de rede disponível. A cada segundo, para cada torre  $t_i$ , são coletadas informações sobre a quantidade de dados trafegados, que são representados por  $D = \{d_1, d_2, \dots, d_z\}$ . Para efeitos de atribuição do peso de cada faixa, o tempo médio gasto para os  $\mathfrak{V}$  veículos atravessarem as  $E$  vias também é coletado e representado por  $S = \{s_1, s_2, \dots, s_m\}$ . A notação usada neste artigo está resumida na Tabela 2.

### 3.2 Processamento de Dados e Roteamento

O cálculo do peso  $w_i$  é feito com base nas informações do tempo  $s_i$  (em segundos) para percorrer a via  $e_i$  e os  $d_i$  bytes transmitidos pela torre  $t_i$  que exerce atuação também sobre a via  $e_i$ . Ambos esses dados sofrem uma normalização *minmax*, como representada na

Table 2: Notação

Símbolo	Descrição
$V$	conjunto de vértices
$E$	conjunto de arestas/vias
$W$	conjunto de pesos para cada aresta/via
$\mathfrak{V}$	conjunto de veículos
$R$	conjunto de rotas para cada veículo
$T$	conjunto de torres eNB
$D$	conjunto de bytes transmitidos por cada torre
$S$	conjunto de tempo para percorrer cada via

Equação 1, resultando na mesma escala  $[0, 1]$  da transmissão ( $sN_i$ ) e tempo de viagem ( $dN_i$ ). Desta forma, eles podem ser manipulados em conjunto estando em medidas iguais, ou seja, a escala de proporção não irá interferir na avaliação do ambiente urbano para o roteamento dos veículos.

$$sN_i = \frac{s_i - s_{min}}{s_{max} - s_{min}} \quad dN_i = \frac{d_i - d_{min}}{d_{max} - d_{min}} \quad (1)$$

Após a normalização dos dados, é atribuído um peso para cada via, que é calculado através da média entre esses valores, utilizando a Equação 2. Para cada via  $e_i$ , o seu peso  $w_i$  será dado pela média aritmética entre os valores normalizados do tempo médio para percorrer-la e da quantidade de bytes  $d_i$  transmitidos pela torre  $t_i$  que exerce atuação sobre essa via.

$$w_i = \frac{(sN_i + dN_i)}{2} \quad (2)$$

A partir do processo de cálculo, todos os pesos/custos das vias estão presentes no conjunto  $W$ . Em seguida, o algoritmo de caminho mais curto, Dijkstra, é usado para gerar a rota para cada veículo em  $v_i$ , tentando reduzir os dois indicadores que influenciam no custo da via.

O algoritmo 1 apresenta os dois processos: (I) o cálculo dos pesos das vias; e, (II) as rotas de cada veículo. Da linha 6 à linha 12, é realizada a atualização dos pesos de cada via. Primeiramente, os

dados são coletados (linhas 7 e 8) e, posteriormente, esses dados são normalizados (linhas 9 e 10). Finalmente, a partir dos dados normalizados, os pesos são atribuídos na linha 11.

---

**Algorithm 1:** Definição da Rotas
 

---

```

1 Entrada:
2  $\mathfrak{V}$   $\leftarrow$  Conjunto de veículos
3  $T$   $\leftarrow$  Conjunto de torres
4  $G$   $\leftarrow$  Grafo representando a rede de vias
5 Ação:
6 foreach  $e_i \in E$  do
7    $s_i \leftarrow \text{getTime}(e_i)$ 
8    $d_i \leftarrow \text{getThroughput}(e_i, T)$ 
9    $sNormalized_i \leftarrow \text{getNormalizedValue}(s_i)$ 
10   $dNormalized_i \leftarrow \text{getNormalizedValue}(d_i)$ 
11   $w_{e_i} \leftarrow \text{getWeight}(sNormalized_i, dNormalized_i)$ 
12 end
13 foreach  $v_i \in \mathfrak{V}$  do
14    $vOrigin \leftarrow \text{getOrigin}(v_i)$ 
15    $vDestination \leftarrow \text{getDestination}(v_i)$ 
16   if  $vOrigin \neq vDestination$  then
17      $r_i \leftarrow \text{getRoute}(vOrigin, vDestination)$ 
18      $\text{updateAdditionalWeight}(G, T, r_i, v_i)$ 
19   end
20 end

```

---

Entre as linhas 13 e 20, é realizado o roteamento dos veículos. As linhas 14 e 15 obtêm as vias de origem e destino para o veículo  $v_i$ , respectivamente. A partir dessas informações, a rota mais curta  $r_i$  é calculada na linha 17 e, na linha 18, um peso adicional é incrementado nas vias que fazem parte da rota gerada (uma vez que a rota dos veículos para essas vias terá mais congestionamento e, possivelmente, transmissão de dados).

### 3.3 Balanceamento de Tráfego

O re-roteamento de veículos pode resultar em um novo problema: o roteamento de muitos veículos para as mesma rotas. Isso faz com que, a longo prazo, outro congestionamento seja criado em vias que anteriormente estavam mais livres para o tráfego. Para evitar esse transtorno, foi utilizado como base a função heurística do algoritmo AR\* [16] para adicionar um fator extra a ser considerado. Este fator extra aumenta com uma força repulsiva o peso/custo das vias selecionadas como parte da nova rota de um veículo, evitando seu uso de forma excessiva.

Na linha 18 do Algoritmo 1, no momento da atribuição de uma rota para um veículo  $v_i$ , os pesos das vias ( $e_i$ ) que fazem parte da nova rota são atualizados para refletir o novo momento. Esse processo de atualização é demonstrado no Algoritmo 2.

No início do processamento do Algoritmo 2, são adquiridas a torre  $t_i$  que cobre a via  $e_i$  na linha 10 e o conjunto  $\bar{E}$  com todas as vias que estão sob essa mesma torre  $t_i$  na linha 11, para que seus pesos sejam atualizados posteriormente. A partir dessas informações, iteramos sobre cada via atualizando seus custos e também marcando

---

**Algorithm 2:** Balanceamento de Tráfego
 

---

```

1 Entrada:
2  $G$   $\leftarrow$  Grafo representando a rede de vias
3  $T$   $\leftarrow$  Conjunto de torres
4  $r_i$   $\leftarrow$  Conjunto de vias de uma rota
5  $v_i$   $\leftarrow$  Veículo que realiza a rota  $r_i$ 
6 Ação:
7  $updated$   $\leftarrow$  Conjunto de vias atualizadas
8 foreach  $e_i \in E$  do
9   if  $e_i \in r_i$  then
10      $t_i \leftarrow \text{getTower}(e_i, T)$ 
11      $\bar{E} \leftarrow \text{getRoadsUnderTower}(t_i)$ 
12     foreach  $[\bar{e}_i, w_i] \in \bar{E}$  do
13       if  $\bar{e}_i \notin updated$  then
14          $updated.add(\bar{e}_i)$ 
15          $\bar{d}_i \leftarrow \text{getBytesFromVehicle}(v_i)$ 
16          $\bar{dN}_i \leftarrow \text{getNormalizedValue}(\bar{d}_i)$ 
17          $\text{addBytesWeight}(w_i, \bar{dNormalized}_i)$ 
18       end
19     end
20   end
21 end

```

---

aquelas que já foram atualizados para evitar atualizar a mesma via novamente.

O Algoritmo 2 atualiza o peso das vias ao adicionar a quantidade de bytes transmitidos pelo dado veículo  $v_i$  em um dos fatores que compõem o custo das vias, sendo este a quantidade de bytes  $d_i$  trafegados pela torre  $t_i$  que exerce cobertura sobre essa via, como exemplificado na linha 15, onde é adquirido a quantidade de bytes transmitidos pelo veículo  $v_i$ , e na linha 17, momento em que essa quantidade de bytes já normalizada é adicionada ao peso  $w_i$ . Esse valor a mais é adicionado não só na via  $e_i$  que será utilizada pelo veículo  $v_i$ , mas também a todas as outras vias que estão sob a cobertura dessa mesma torre  $t_i$ .

Um exemplo do contexto deste artigo é ilustrado na Figura 1, onde um ambiente urbano é apresentado e as vias nele possuem um nível de congestionamento de tráfego singular. Além disso, é possível notar que cada via é coberta por uma ou mais torres LTE e as torres possuem uma disponibilidade de recursos de rede específica.

## 4 EXPERIMENTOS

Esta seção apresenta as configurações do cenário usado nos experimentos (ambiente de simulação na Seção 4.1 e configuração do cenário na seção 4.2) e os resultados obtidos (Seção 5). O desempenho do DINO foi comparado com o algoritmo Dynamic Shortest Path (DSP) [16], que atribui aos veículos o caminho com o menor tempo de viagem.

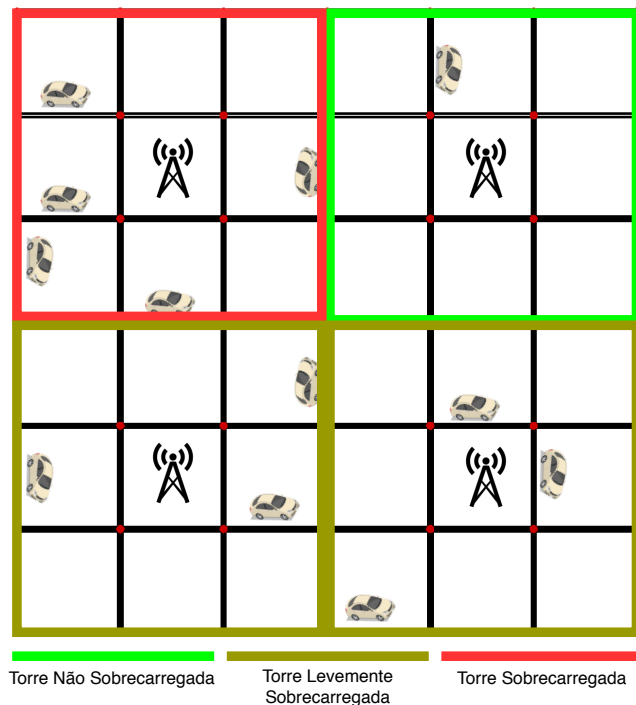


Figure 1: Ilustração do cenário com quatro torres eNB.

#### 4.1 Ambiente de Simulação

Nós empregamos SUMO<sup>1</sup>, TraCI API<sup>2</sup> e NS-3<sup>3</sup> para nossas simulações. SUMO é um pacote de simulação microscópica de tráfego rodoviário de código aberto altamente portátil, projetado para lidar com grandes redes rodoviárias. TraCI é uma biblioteca que fornece comandos abrangentes para controlar o comportamento da simulação do SUMO, incluindo o status do veículo e a configuração de estradas e semáforos. O NS-3 é um simulador de rede de eventos discretos para sistemas de internet, cujo objetivo principal é ser um ambiente de simulação aberto para uso em pesquisa e ensino de redes.

A estratégia de roteamento foi implementada usando TraCI em conjunto com o NS-3. As funcionalidades da API foram utilizadas na simulação da rede LTE por meio de modificações feitas em um módulo de acoplamento bidirecional entre os simuladores de mobilidade e de rede, que está disponível online<sup>4</sup>. Como a simulação de mobilidade no SUMO é controlada pelo módulo de rede NS-3 LTE por meio do TraCI, a cada intervalo de tempo médio para percorrer as estradas, novas rotas são calculadas e atribuídas aos veículos.

#### 4.2 Configuração de Cenário

Para a rede de vias urbanas, foi utilizado o formato de grade do Modelo de Mobilidade Manhattan [17]. Nesta topologia, os veículos podem viajar em ambas as direções em cada via. Além disso, os veículos precisam escolher a direção de conversão (esquerda ou

direita) ou seguir em frente em cada interseção. A probabilidade de seguir em frente é de 50% e, virar à esquerda ou à direita, é de 25% cada. No cenário experimental, o tráfego de veículos é gerado aleatoriamente.

Definimos dois tamanhos de mapas diferentes, mas mantendo uma relação entre o tamanho dos mapas e o número de torres eNB, de forma que as torres sempre ocupem uma área de operação de 40000m<sup>2</sup> and 90000m<sup>2</sup>. Para o cenário com quatro torres, foram criados dois cenários: (1) 400m x 400m com 80 veículos e (2) 600m x 600m com 160 veículos. Para o cenário com nove torres, outros dois cenários foram definidos: (i) 600m x 600m com 200 veículos e (ii) 900m x 900m com 300 veículos. Estes 4 cenários foram definidos para fornecer uma avaliação do DINO em situações distintas em cenários urbanos.

Em relação as aplicações, três configurações foram implantadas: Mensagens CAM, VoIP e *streaming* de vídeo. Essas aplicações são comumente usadas em redes LTE [5, 12]. As informações sobre os parâmetros das aplicações e a configuração da rede LTE (torres e dispositivos) estão detalhadas na Tabela 3.

Table 3: Parâmetros dos Experimentos

Rede LTE	Parâmetro	Valor
	eNodeB Potência Tx	46 dBm
	UE Potência Tx	10 dBm
Aplicação CAM	Parâmetro	Valor
	Tamanho dos Pacotes	800 Bytes
	Intervalo entre Pacotes	100 ms
VoIP	Parâmetro	Valor
	Tamanho dos Pacotes	40 Bytes
	Intervalo entre Pacotes	20 ms
Video Streaming	Parâmetro	Valor
	Tamanho dos Pacotes	1000 Bytes
	Intervalo entre Pacotes	75 ms

## 5 RESULTADOS

Esta subseção apresenta os resultados dos experimentos realizados, comparando o desempenho da proposta DINO com o algoritmo DSP para TM em ITS. Portanto, avaliamos o impacto do DINO no tempo de viagem e na entrega dos pacotes das aplicações.

Durante os experimentos, as seguintes métricas foram analisadas: (a) Taxa de Entrega de Pacotes de Dados, a porcentagem de pacotes de dados entregues com sucesso. (b) Tempo de Viagem (em segundos), tempo médio gasto pelos veículos para chegar ao destino a partir da posição de origem; e, (c) Distância Percorrida (em metros), distância média percorrida pelos veículos para chegar ao destino a partir da posição de origem.

As taxas de entrega de pacotes para os quatro cenários são ilustradas na Figura 2. O Cenário 1 tem a menor carga, uma vez que há menos veículos no cenário. Esse fato resultou em uma taxa de entrega muito alta de ambas as abordagens, mas o DINO superou o DSP em 7% (em média). Por outro lado, no caso de maior carga (Cenário 4) o desempenho dos algoritmos ficam mais próximos, onde o DINO teve uma taxa de entrega 3% maior. Esse desempenho mais próximo ocorre devido ao fato do número de veículos ser elevado em relação aos recursos disponíveis da rede nas torres LTE, limitando a capacidade do DINO de balancear as rotas no meio

<sup>1</sup>[eclipse.org/sumo/](http://eclipse.org/sumo/)

<sup>2</sup>[sumo.dlr.de/docs/TraCI/C++TraCI.html](http://sumo.dlr.de/docs/TraCI/C++TraCI.html)

<sup>3</sup>[nslam.org/releases/ns-3-30/](http://nslam.org/releases/ns-3-30/)

<sup>4</sup><https://github.com/vodafone-chair/ns3-sumo-coupling>

urbano. A proporção de veículos por torre é de 20 no Cenário 1, enquanto no Cenário 4 é mais de 33.

Em relação aos Cenários 2 e 3, apesar de suas proporções distintas (40 no Cenário 2 e 22 no Cenário 3), o desempenho do DINO é semelhante. Isso ocorre devido à capacidade do DINO em balancear o tráfego nas rotas pelo meio urbano, potencializando a QoS das aplicações. O cenário 3 destaca a eficiência da definição de rota do DINO, caso esse em que o desempenho do DINO chega a ser 10% melhor que o DSP.

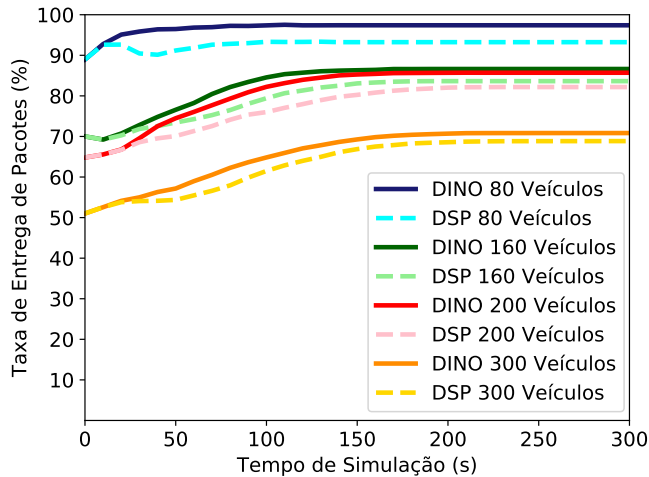


Figure 2: Taxa de entrega de pacotes em Quatro Cenários: (1) 4 Torres com 80 veículos, (2) 4 Torres com 160 veículos, (3) 9 Torres com 200 veículos, e (4) 9 Torres com 300 veículos.

Os resultados relacionados ao tráfego de veículos são apresentados nas Figuras 3 e 4.

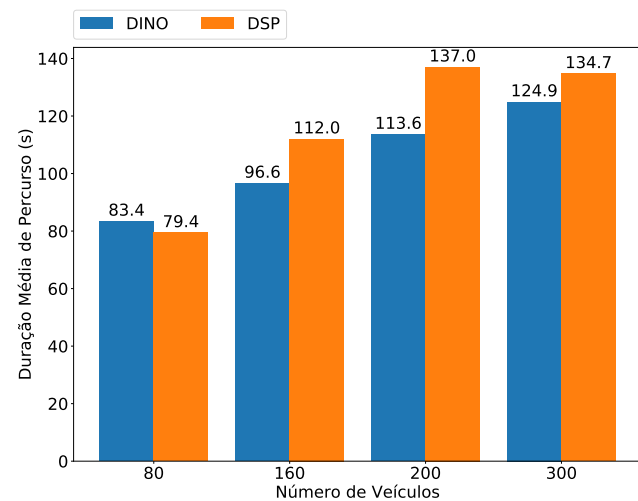


Figure 3: Tempo de Viagem.

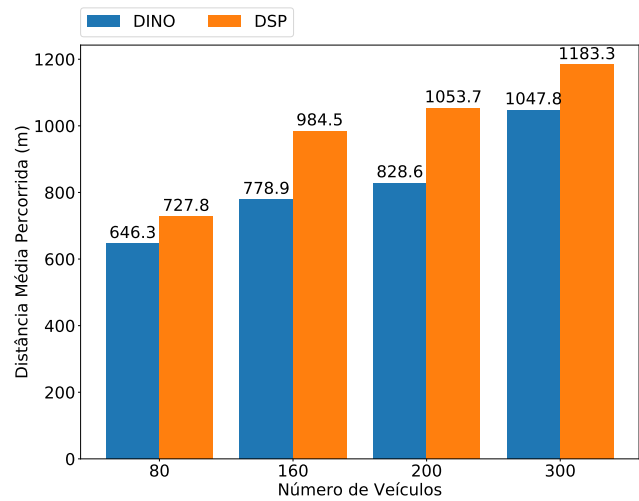


Figure 4: Distância Percorrida.

Em todos os cenários, a distância percorrida e os indicadores de duração da viagem de DINO são menores do que DSP. Este fato demonstra que o DINO teve sucesso não só na redução do congestionamento nas torres eNB, mas também nas próprias estradas, equilibrando as rotas com base nas rotas previamente calculadas.

Como conclusão, é possível notar que o DINO melhora a taxa de entrega de pacotes dos veículos, enquanto o tempo de deslocamento é reduzido quando comparado à abordagem existente.

## 6 CONCLUSÃO

Os ambientes urbanos das cidades modernas sofrem com o engarrafamento e congestionamento das estradas. Uma consequência dessa situação é o aumento do tempo de viagem dos veículos. Uma abordagem adequada para mitigar esses problemas é usar um ITS com uma estratégia de gerenciamento de tráfego. Adicionalmente, o ITS deve estar atento aos recursos de rede que suportam a comunicação dos veículos e suas aplicações.

Neste artigo, propomos uma estratégia para definir rotas e realizar o re-roteamento de veículos em um TM de ITS. A proposta, denominada DINO, visa evitar o esgotamento dos recursos de rede das torres LTE e minimizar o tempo de viagem dos veículo no ambiente urbano. De acordo com os experimentos realizados, o DINO reduz tanto a perda de pacotes das aplicações quanto o tempo de viagem quando comparado à abordagem existente.

Como trabalhos futuros, pretendemos aplicar uma abordagem multi-objetivo para incluir o reconhecimento de múltiplas tecnologias de comunicação na estratégia de roteamento e re-roteamento. Além disso, incluiremos a avaliação da proposta em distintos cenários urbanos.

## REFERÊNCIAS

- [1] George Dimitrakopoulos and Panagiotis Demestichas. Intelligent transportation systems. *IEEE Vehicular Technology Magazine*, 5(1):77–84, 2010.
- [2] Daniel Jiang and Luca Delgrossi. Ieee 802.11 p: Towards an international standard for wireless access in vehicular environments. In *VTC Spring 2008-IEEE Vehicular Technology Conference*, pages 2036–2040. IEEE, 2008.

- [3] Marica Amadeo, Claudia Campolo, and Antonella Molinaro. Enhancing ieee 802.11 p/wave to provide infotainment applications in vanets. *Ad Hoc Networks*, 10(2):253–269, 2012.
- [4] Giuseppe Araniti, Claudia Campolo, Massimo Condoluci, Antonio Iera, and Antonella Molinaro. Lte for vehicular networking: a survey. *IEEE communications magazine*, 51(5):148–157, 2013.
- [5] Izhak Rubin, Andrea Baiocchi, Yulia Sunyoto, and Ion Turcanu. Traffic management and networking for autonomous vehicular highway systems. *Ad Hoc Networks*, 83:125 – 148, 2019. ISSN 1570-8705. doi: <https://doi.org/10.1016/j.adhoc.2018.08.018>. URL <http://www.sciencedirect.com/science/article/pii/S1570870518306164>.
- [6] Hanbyul Seo, Ki-Dong Lee, Shinpei Yasukawa, Ying Peng, and Philippe Sartori. Lte evolution for vehicle-to-everything services. *IEEE communications magazine*, 54(6):22–28, 2016.
- [7] D. L. Guidoni, G. Maia, F. S. H. Souza, L. A. Villas, and A. A. F. Loureiro. Vehicular traffic management based on traffic engineering for vehicular ad hoc networks. *IEEE Access*, 8:45167–45183, 2020.
- [8] A. T. Akabane, R. L. Gomes, R. W. Pazzi, E. R. M. Madeira, and L. A. Villas. Apolo: A mobility pattern analysis approach to improve urban mobility. In *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, pages 1–6, 2017.
- [9] Artur Pessoa, Ruslan Sadykov, Eduardo Uchoa, and François Vanderbeck. A generic exact solver for vehicle routing and related problems. *Mathematical Programming*, pages 1–41, 2020.
- [10] Efraim Ruiz, Valeria Soto-Mendoza, Alvaro Ernesto Ruiz Barbosa, and Ricardo Reyes. Solving the open vehicle routing problem with capacity and distance constraints with a biased random key genetic algorithm. *Computers Industrial Engineering*, 133:207 – 219, 2019. ISSN 0360-8352. doi: <https://doi.org/10.1016/j.cie.2019.05.002>. URL <http://www.sciencedirect.com/science/article/pii/S0360835219302694>.
- [11] Allan M de Souza, Roberto S Yokoyama, Guilherme Maia, Antonio Loureiro, and Leandro Villas. Real-time path planning to prevent traffic jam through an intelligent transportation system. In *2016 IEEE Symposium on Computers and Communication (ISCC)*, pages 726–731. IEEE, 2016.
- [12] Z. Ning, X. Wang, J. J. P. C. Rodrigues, and F. Xia. Joint computation offloading, power allocation, and channel assignment for 5g-enabled traffic management systems. *IEEE Transactions on Industrial Informatics*, 15(5):3058–3067, 2019.
- [13] Eric J Horvitz, Johnson Apacible, Raman Sarin, and Lin Liao. Prediction, expectation, and surprise: Methods, designs, and study of a deployed traffic forecasting service. *arXiv preprint arXiv:1207.1352*, 2012.
- [14] ETSI. Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service. TS 102 637-2 V1.1.1, ETSI, 4 2010.
- [15] Alexey Vinel. 3gpp lte versus ieee 802.11 p/wave: Which technology is able to support cooperative vehicular safety applications? *IEEE Wireless Communications Letters*, 1(2):125–128, 2012.
- [16] Juan Pan, Iulian Sandu Popa, Karine Zeitouni, and Cristian Borcea. Proactive vehicular traffic rerouting for lower travel time. *IEEE Transactions on vehicular technology*, 62(8):3551–3568, 2013.
- [17] Atulya Mahajan, Niranjana Potnis, Kartik Gopalan, and Andy Wang. Urban mobility models for vanets. In *2nd IEEE International Workshop on Next Generation Wireless Networks*, volume 33, page 38, 2006.