

**POR AQUI MAIS SEGURO: MELHORANDO A MOBILIDADE
E A SEGURANÇA NAS VIAS URBANAS**

DIXIM

Por Aqui é Mais Seguro: Melhorando a Mobilidade e a Segurança nas Vias Urbanas

Allan M. de Souza¹, Leonardo C. Botega^{2,3}, Islene Calciolari Garcia¹, Leandro A. Villas¹

¹Instituto de Computação – UNICAMP

²Centro Universitário Eurípides de Marília – UNIVEM

³Strategic Intelligence – Stratelli

Abstract. *Recently, many cities are facing mobility and safety issues, commonly related to traffic congestion and to elevated number of city-wide criminal incidents. Several Intelligent Transportation Systems (ITS) were proposed to overcome mobility issues; meanwhile, some safety-based systems were proposed to guide pedestrians and drivers toward safest paths. However, most of these systems tackle only one of the issues. Hence, an ITS can guide vehicles toward risky areas, in order to avoid traffic congestion, while a safety-based system can guide them toward congested roads, focusing on the safety of drivers and passengers. This paper introduces #PAS (Por Aqui é mais Seguro), a hybrid ITS which employs accurate knowledge about traffic conditions and unsafety levels on roads for improving the safety of drivers and passengers at the same time it deals with traffic congestion. Simulation results under a realistic scenario have shown that #PAS outperformed state-of-the-art approaches that deal with mobility or safety issues.*

Resumo. *Recentemente, muitas cidades estão enfrentando problemas de mobilidade e segurança, comumente relacionados aos congestionamentos e aos índices elevados de incidentes criminais. Muitos sistemas de transporte inteligentes (ITSs) foram desenvolvidos para tratar o problema de mobilidade urbana, ao mesmo tempo, alguns sistemas baseados em segurança foram criados para direcionar veículos e pedestres por rotas mais seguras. Entretanto, a maioria desses sistemas consideram apenas um problema. Portanto, um ITS pode direcionar veículos por ruas perigosas, a fim de evitar congestionamentos, enquanto sistemas baseados em segurança podem guiar veículos para vias congestionadas visando melhorar a segurança dos motoristas e passageiros. Com isso em mente, nesse trabalho propomos #PAS (Por Aqui é mais Seguro), um ITS híbrido que possui conhecimento sobre as condições de tráfego e também sobre os níveis de insegurança nas vias para melhorar a mobilidade urbana e também a segurança dos motoristas e passageiros. Os resultados em um cenário realista mostraram que o #PAS supera as soluções do estado-da-arte para tratar os problemas de mobilidade e segurança nas vias urbanas.*

1. Introdução

Congestionamento de veículos é um problema diário para habitantes de grandes centros urbanos, pois é comumente associado a previsões incorretas para o tempo de viagem, problemas de saúde, impactos negativos no desenvolvimento econômico e também no meio-ambiente [Djahel et al. 2015, de Souza et al. 2017b]. Diante desse cenário, os habitantes de grandes centros urbanos recorrem a sistemas de navegação veicular (VNS — *Vehicular Navigation Systems*) para lidar com os problemas produzidos pelos congestionamentos.

Google Maps, TomTom and WAZE são VNSs existentes que sugerem rotas mais rápidas para os usuários a partir de conhecimento de tráfego global [Garip et al. 2015]. Entretanto, esses sistemas não consideram questões inerentes à segurança dos motoristas e passageiros,

podendo guiá-los por ruas e bairros perigosos, conseqüentemente gerando situações de risco incluindo roubo, sequestro e até mesmo assassinatos [(CNN) 2015, Online 2016]. Por exemplo, um casal foi baleado e a mulher veio a falecer em uma favela no Rio de Janeiro depois de serem guiados pelo WAZE [(CNN) 2015]. Um outro exemplo, é o registro mostrado em [Online 2016], que mostra um veículo cruzando um tiroteio em Boston, nos Estados Unidos, também guiado por um VNS visando melhorar a mobilidade urbana.

Uma solução viável para essas limitações é a utilização de sistemas de transporte inteligentes, os quais integram diferentes sistemas para sensoriamento, processamento e comunicação a fim de prover um trânsito mais eficiente e seguro [Djahel et al. 2015]. Para isso, os ITSs utilizam as redes veiculares (VANETs — *Vehicular Ad hoc Networks*) para que os veículos possam compartilhar e receber informações de outros veículos, infraestruturas, *smart devices*, subsistemas e até mesmo pessoas [de Souza et al. 2017b]. Isso permite que os ITSs criem um conhecimento sobre diversas variáveis de um ambiente urbano, incluindo, mobilidade, insegurança, pontos de interesse, níveis de poluição e etc, conseqüentemente, permitindo que os ITSs derivem soluções mais inteligentes para a recomendação de rota, como por exemplo, sugerir uma rota mais rápida e também mais segura. Contudo, muitos ITSs encontrados na literatura criam um conhecimento sobre as condições de tráfego para a recomendação de rotas [Doolan and Muntean 2017, de Souza et al. 2016b, de Souza et al. 2017a, de Souza et al. 2016a, Wang et al. 2015], naturalmente possuindo os mesmos problemas de insegurança presentes nos VNSs.

Visando prover uma recomendação de rota segura, sistemas baseados no sensoriamento participativo foram propostos [Shah et al. 2011]. Porém, esses sistemas apresentam limitações relacionadas a: (i) dependência de relatos dos usuários; (ii) informações de qualidade limitada, incluindo dados incompletos, desatualizados, inconsistentes ou falsos; e (iii) falta de conhecimento sobre a mobilidade urbana. Como resultado, sistemas puramente baseados em sensoriamento participativo podem emergir vários problemas, incluindo a produção de falsos positivos sobre pontos de insegurança. Para superar este problema, alguns sistemas que também focam na recomendação de rota segura, utilizam informações oficiais de base de dados dos departamentos de polícia para construir conhecimento sobre pontos de insegurança [Ballesteros et al. 2012]. Entretanto, devido à falta de conhecimento sobre as condições de tráfego, esses sistemas podem criar congestionamentos em vários pontos, potencialmente motivando crimes nessas áreas, como por exemplo roubo de veículos. Sendo assim, sugerir rotas seguras e melhorar a eficiência do tráfego de veículos na cidade ainda é uma questão em aberto.

Apesar das soluções [Shah et al. 2011, Galbrun et al. 2016, Santos et al. 2017, de Souza et al. 2016a] minimizarem os problemas de mobilidade e de insegurança, recomendando rotas mais rápidas ou mais seguras, elas empregam uma arquitetura centralizada, assim, gerando problemas de contenção do canal de comunicação e escalabilidade na recomendação de rotas dependendo da densidade de veículos.

Motivados pelas limitações apresentadas nas soluções existentes, apresentamos #PAS (Por Aqui é mais Seguro), um ITS híbrido para melhorar a mobilidade e segurança nas vias urbanas. #PAS constrói conhecimentos sobre: (i) condições de tráfego, baseado em informações compartilhadas pelos veículos, e (ii) áreas inseguras, utilizando informações processadas à partir de dados oficiais dos boletins de ocorrência das bases de dados dos departamentos de polícia [Botega et al. 2017]. Além disso, #PAS modela o problema de melhorar a mobilidade e a segurança como uma instância do problema de caminhos mínimos com restrição de recurso (RCSP - *Resource Constrained Shortest Path*) [Jaffe 1984], o qual é um problema é NP-Difícil

e admite um algoritmo pseudo-polinomial baseado em programação dinâmica [Joksch 1966]. É importante salientar que o tamanho da tabela de programação dinâmica depende linearmente do valor máximo de insegurança, o que a torna viável para o #PAS uma vez que o valor máximo de insegurança não é arbitrariamente grande. Por fim, #PAS possui uma arquitetura híbrida para permitir o cálculo das rotas em tempo real e também minimizar as contenções na rede.

Os resultados das simulações mostraram que quando comparado com outras abordagens que utilizam apenas uma informação (i.e., mobilidade ou segurança), as rotas calculadas pelo #PAS possuem uma boa relação entre segurança e eficiência do tráfego, onde é possível reduzir o tempo de viagem em 60% dos veículos e aumentar a segurança em 90% dos veículos. Além disso, #PAS ainda reduz a quantidade de mensagens transmitidas em 98%.

O restante do trabalho é organizado da seguinte forma. Seção 2 descreve os trabalhos relacionados, apresentando vantagens e limitações. A solução proposta é descrita na Seção 3. A análise de desempenho da solução proposta é apresentada na Seção 4. Por fim, a Seção 5 apresenta as conclusões e indica trabalhos futuros.

2. Trabalhos Relacionados

Com o intuito de melhorar a mobilidade urbana, VNSs frequentemente recomendam rotas mais rápidas baseadas em um conhecimento global, sempre que uma requisição de rota é efetuada. Entretanto, VNSs não possuem um planejamento de rotas dinâmico e também não lidam com eventos inesperados em rota. Por outro lado, ITSs solucionam essas limitações empregando mecanismos para planejamento de rotas dinâmico [Doolan and Muntean 2017, Pan et al. 2016, de Souza et al. 2017a, Wang et al. 2015] e para detectar eventos inesperados [de Souza et al. 2016a, Wang et al. 2016].

Contudo, tanto os VNSs quanto os ITSs atuais apenas consideram informações de tráfego para a recomendação de rotas, eles podem direcionar veículos para ruas e bairros perigosos a fim de evitar congestionamentos, consequentemente colocando a vida dos motoristas e passageiros em risco. Diante desse cenário, para evitar esse problema soluções baseadas em recomendação de rotas seguras foram propostas [Shah et al. 2011, Ballesteros et al. 2012, Galbrun et al. 2016, Santos et al. 2017].

Shah et al. [Shah et al. 2011] introduziram o CROWDSAFE, um sistema baseado em *crowdsourcing* para sugerir rotas mais seguras para os motoristas. O sistema permite que os usuários reportem incidentes criminais para um servidor central utilizando *smartphones*, em seguida, processando os incidentes recebidos, o sistema constrói um conhecimento sobre áreas inseguras. Dessa forma, CROWDSAFE é capaz de sugerir rotas mais seguras para os usuários, mas, como o sistema é puramente *crowdsourcing*, o conhecimento sobre áreas inseguras pode ser limitado ou incorreto, pois muitos crimes podem não ser reportados ou ser até mesmo falsos. Além disso, como o sistema não possui o conhecimento sobre as condições de tráfego, as rotas recomendadas podem piorar ainda mais a mobilidade, uma vez o sistema pode criar congestionamentos em áreas com baixa insegurança e também direcionar veículos por ruas congestionadas.

Com uma abordagem diferente de [Shah et al. 2011], Galbrun et al. [Galbrun et al. 2016] apresentaram um sistema de recomendação de rotas seguras baseado em informações de crimes oficiais, disponíveis por uma iniciativa do governo dos Estados Unidos¹. Outro diferencial é que o sistema também é ciente da distância entre origem e destino para a recomendação de rotas, assim, o sistema pode recomendar rotas menores

¹<http://www.data.gov/about>

(em distância) evitando áreas de risco. Para isso, o sistema calcula todas as possíveis rotas entre origem e destino e também calcula a distância total e insegurança de cada rota, assim podendo recomendar a melhor rota em distância e segurança. Entretanto, calcular todas as rotas possíveis pode ser muito custoso, consequentemente degradando a eficiência do sistema. Adicionalmente, utilizar apenas informações sobre a distância não oferece uma melhor mobilidade, devido a falta de conhecimento sobre as condições de tráfego das vias. Portanto, de forma semelhante ao CROWDSAFE, os veículos podem ser direcionados por vias congestionadas para evitar áreas inseguras, consequentemente piorando a mobilidade.

É importante salientar que apenas as soluções [Doolan and Muntean 2017, de Souza et al. 2017a, Pan et al. 2016, de Souza et al. 2016b], são escaláveis e possibilitam o cálculo de rotas em tempo real, pois possuem uma arquitetura distribuída. Soluções com arquitetura centralizada [Shah et al. 2011, Galbrun et al. 2016, Santos et al. 2017, de Souza et al. 2016a] podem introduzir uma latência indesejável para o sistema dependendo do número de requisições, assim, afetando a performance geral do sistema.

As soluções apresentadas possuem limitações que incluem problemas de insegurança e escalabilidade, ausência de um mecanismo para recomendação de rota em tempo real, recomendações de rotas ineficientes em relação mobilidade e segurança, falta de conhecimento sobre condições de tráfego e áreas inseguras. Sendo assim, para minimizar essas limitações apresentadas, apresentamos #PAS, um ITS híbrido para cálculo de rota em tempo real para melhorar a mobilidade e segurança nas vias urbanas.

3. #PAS: Por Aqui é mais Seguro

Os principais desafios em desenvolver um ITS para melhorar a mobilidade e a segurança nas vias urbanas são: (i) como construir um conhecimento preciso sobre as condições de tráfego e áreas inseguras; e (ii) como direcionar veículos por rotas mais seguras sem degradar a eficiência do tráfego. Portanto, o sistema deve possuir informações precisas sobre áreas de insegurança e também sobre o comportamento do trânsito, para não direcionar veículos por áreas perigosas a fim de evitar congestionamentos, e também para não criar gargalos na infraestrutura de transporte por evitar ruas e bairros perigosos.

Definição 1: *Seja $D = (V, E)$ um digrafo representando o cenário, onde o conjunto de vértices V representa as interseções, e o conjunto de arestas $E \subseteq V \times V$ corresponde às vias, i.e., uma aresta $uv \in E$ corresponde à via que conecta as interseções u e v , para $u, v \in V$. Cada aresta $uv \in E$ possui dois atributos, $\tau_{uv} \geq 0$ and $\mu_{uv} \geq 0$ os quais representam as condições de tráfego e a insegurança da via uv , respectivamente. Além disso, considere N como sendo o conjunto de veículos em circulação no cenário, onde cada veículo $v \in N$ possui um par de origem $s \in V$ e destino $t \in V$, tal que $s \neq t$, e é associado uma rota $P \subseteq E$ conectando s a t . Por fim, a condição de tráfego da rota de cada veículo é definida por $\tau_P = \sum_{uv \in P} \tau_{uv}$ e sua insegurança por $\mu_P = \sum_{uv \in P} \mu_{uv}$.*

3.1. Visão geral do sistema

A Figura 1(a) apresenta uma visão geral sobre o funcionamento do sistema. #PAS emprega uma arquitetura híbrida, composta por um servidor e um software executado na OBU (*On Board Unit*) de cada veículo participante do sistema. Os veículos se comunicam com o servidor utilizando tanto comunicações veículo para veículo (V2V — *Vehicle-to-Vehicle*) quanto comunicações veículo para infraestrutura (V2I — *Vehicle-to-Infrastructure*) para reportarem suas informações sobre o tráfego (periodicamente atualizado pelas informações dos veículos). Dessa forma, o servidor é capaz de construir um conhecimento preciso sobre as condições de tráfego. Além disso, o servidor possui informações sobre eventos criminais (i.e., boletins de

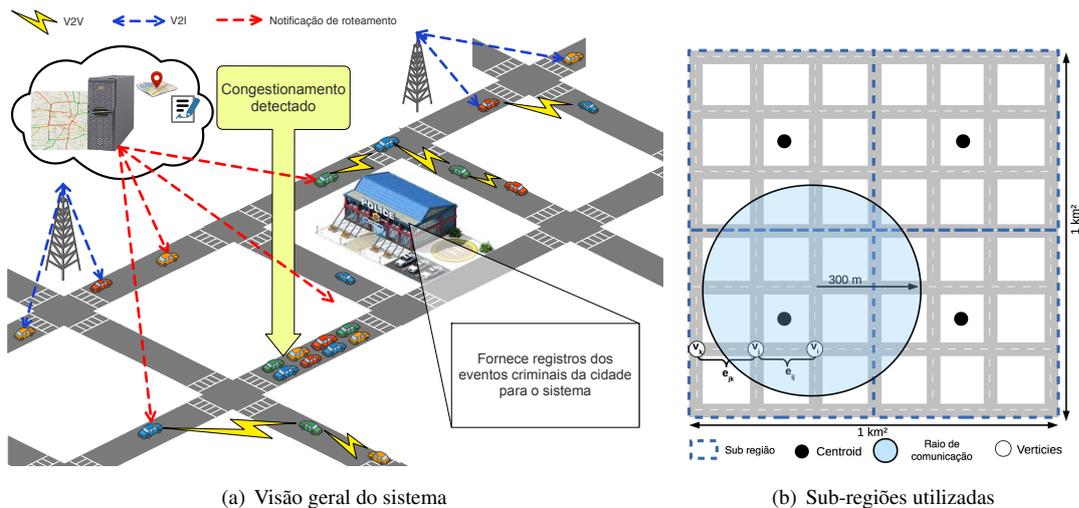


Figura 1. Visão geral do funcionamento do #PAS incluindo sub-regiões utilizadas

ocorrência registrados pelos departamentos de polícia) processados, fornecidos pela Secretaria de Segurança Pública do Estado de SP (SSP-SP) para criar um conhecimento sobre os níveis de insegurança nas vias.

Após criar o conhecimento sobre as condições de tráfego, o servidor é capaz de detectar vias congestionadas, e assim que um congestionamento é detectado, o servidor envia uma notificação de roteamento para os veículos que irão passar pelo congestionamento. Quando um veículo próximo ao congestionamento recebe uma notificação do servidor, ele a dissemina para seus vizinhos para que eles também sejam alertados sobre o congestionamento. A notificação de roteamento contém os conhecimentos sobre condições de tráfego e áreas inseguras construídos pelo servidor, assim, permitindo que os próprios veículos calculem uma nova rota para melhorar sua mobilidade e segurança.

3.2. Reportando informações de tráfego

Para criar um conhecimento preciso sobre as condições de tráfego, o servidor precisa receber informações de todos os veículos nas vias. Entretanto, se todos os veículos reportarem suas informações periodicamente, um grande número de transmissões seria gerado, potencialmente sobrecarregando o canal de comunicação e, conseqüentemente, degradando a eficiência geral do sistema. Sendo assim, #PAS emprega um mecanismo eficiente para reduzir o número de mensagens transmitidas e ainda manter um conhecimento preciso sobre as condições de tráfego. Para isso, #PAS utiliza a abordagem apresentada em [de Souza and Villas 2016], onde o cenário é dividido em sub-regiões baseado no tamanho do raio de comunicação dos veículos, além disso, para cada sub-região é definido um *centroid*. A Figura 1(b) exemplifica a divisão do cenário.

Todos os veículos em uma mesma sub-região compartilham as suas informações e de seus vizinhos de um salto através de *beacons* periodicamente enviados. Essas informações são: (i) posição atual $wv \in E$; e (ii) velocidade v_n . Dessa forma, todos os veículos em uma mesma sub-região possuem conhecimento da quantidade veículos em cada via e a velocidade de cada um, criando assim um conhecimento da sub-região [de Souza and Villas 2016]. Por fim, para reduzir o número de transmissões, #PAS emprega uma abordagem orientada a atraso, em que todos os veículos em uma mesma sub-região agendam o envio de seu conhecimento previamente gerado baseado na distância entre o veículo e o *centroid* da sua sub-região. Esse mecanismo prioriza os veículos mais próximos de cada *centroid* e, após a transmissão

do conhecimento do primeiro veículo, todos os veículos que possuem o mesmo conhecimento agendado cancelam a sua transmissão, consequentemente reduzindo o número de transmissões.

3.3. Criando conhecimento sobre condições de tráfego

A condição de tráfego de cada via é calculada a partir do conhecimento das sub-regiões enviado pelos veículos. Primeiramente, #PAS calcula a velocidade média de cada via uv a partir das informações sobre quantidade de veículos em uma mesma via e suas respectivas velocidades v_n . A velocidade média de cada via é representada por $vmedia_{uv}$, calculada como:

$$vmedia_{uv} = \frac{\sum_{n \in N_{uv}} v_n}{|N_{uv}|}, \quad (1)$$

onde N_{uv} é o conjunto de veículos na via uv . Após calcular a velocidade média de cada via, #PAS calcula a condição de tráfego τ_{uv} da seguinte forma:

$$\tau_{uv} = \frac{vmax_{uv} - vmedia_{uv}}{vmax_{uv}}, \quad (2)$$

onde $vmax_{uv}$ é a velocidade máxima da via uv . Por questões de clareza, quando o sistema não recebe informações sobre uma determinada via, o sistema define essa via como livre, assim, atribuindo sua velocidade média como o valor da velocidade máxima permitido na via em questão.

Com base nos valores de τ_{uv} é possível detectar gargalos na eficiência do tráfego utilizando a classificação dos níveis de serviço (LOS - *Level-Of-Service*) disponível no manual de capacidade das rodovias (HCM - *Highway Capacity Manual*) [Board 2010]. A Tabela 1 apresenta essa classificação.

Tabela 1. Classificação de τ_{uv}

τ_{uv}	LOS	Classificação do tráfego
(0, 0.15]	A	Fluxo livre
(0.15, 0.33]	B	Fluxo livre
(0.33, 0.50]	C	Congestionamento leve
(0.50, 0.60]	D	Congestionamento leve
(0.60, 0.70]	E	Congestionado
(0.70, 1.00]	F	Congestionado

Tabela 2. Classificação dos eventos criminais (EC)

Tipo	Descrição do crime	np
1	Crimes de baixa complexidade, sem armas de fogo executados por um único indivíduo	1
2	Crimes de complexidade média, executados por dois indivíduos com armas de fogo através de um esquema planejado	2
3	Crimes de alta complexidade, executados por muitos indivíduos com armas de fogo e/ou especiais	4

3.4. Criando conhecimento sobre níveis de insegurança nas vias

A insegurança de cada via é obtida processando as informações dos boletins de ocorrência fornecidos pelos departamentos de Polícia. O sistema aplica mecanismos para: (i) extrair informações do local da ocorrência de cada crime a partir de seu endereço (geocodificação); (ii) mapear a localização de cada crime para a via mais próxima utilizando *geocoding* para obter uma coordenada geográfica (i.e., lat/long); e (iii) caracterizar o quão perigoso é cada crime a partir de seu tipo e complexidade (i.e., roubo, assalto, sequestro, etc). Dessa forma, é possível classificar cada crime e atribuir-lhe um peso baseado em suas características [Botega et al. 2017]. A Tabela 2 mostra como cada crime pode ser classificado, apresentando seu tipo, uma breve descrição e seu nível de perigo (np).

Para mapear os crimes para a via mais próxima, #PAS utiliza a coordenada obtida baseada no endereço extraído do boletim de ocorrência e também as coordenadas dos vértices do cenário. Dessa forma, é possível identificar os vértices $u, v \in V$ mais próximos da coordenada do evento criminal, tal que $uv \in E$.

Após classificar cada crime e mapeá-lo para a sua via mais próxima, #PAS calcula a insegurança de cada via μ_{uv} da seguinte forma:

$$\mu_{uv} = \sum_{c \in EC_{uv}} np_c, \quad (3)$$

onde EC_{uv} é o conjunto de eventos criminais associados à via uv e np_c é o nível de perigo de cada crime c calculado pelo sistema.

3.5. Roteamento distribuído para melhorar a mobilidade e a segurança

Por possuir uma arquitetura híbrida, #PAS permite que os próprios veículos otimizem a mobilidade e a segurança de suas rotas. Portanto, quando o servidor detecta um congestionamento (i.e., $\tau_{uv} \geq 0.6$), ele envia uma mensagem com os conhecimentos sobre as condições de tráfego e insegurança atualizados para o veículo mais próximo do congestionamento que enviou o conhecimento de sua sub-região. A atualização dos conhecimentos enviados possuem apenas as vias com valores τ_{uv} e μ_{uv} diferentes da última atualização.

Após receber a mensagem de atualização dos conhecimentos, o veículo é responsável por disseminá-la para seus vizinhos para que eles tenham conhecimento sobre o congestionamento. Para garantir que os veículos que irão ser afetados pelo congestionamento recebam a mensagem, #PAS aplica um protocolo de disseminação de múltiplos saltos (o número máximo de saltos é parametrizado). Entretanto, para evitar o problema de tempestade de *broadcast*, onde muitos veículos tentam transmitir ao mesmo tempo, #PAS emprega um mecanismo de supressão de *broadcast* baseado em atraso, onde o atraso para retransmissão é baseado na distância inversa entre transmissor e receptor descrito em [de Souza et al. 2016a].

Sendo assim, ao receber a atualização de conhecimento o veículo pode calcular uma nova rota para otimizar sua mobilidade e segurança. Para isso, modelamos esse problema como uma instância do problema de caminhos mínimos com restrição de recurso (RCSP) [Jaffe 1984]. A ideia é encontrar a rota mais rápida cuja insegurança máxima da rota não seja maior que um limitante λ (i.e., o recurso). Formalmente, considere um veículo n com posição atual s e destino t , lembrando que τ_P é a condição de tráfego da rota P conectando s a t , e μ_P é a insegurança de P . Portanto, o problema de encontrar a rota mais segura consiste em:

$$\min \{ \tau_P \mid P \text{ é uma rota de } s \text{ a } t \text{ e } \mu_P \leq \lambda \}. \quad (4)$$

Como a insegurança de cada via pode apenas assumir valores inteiros, a insegurança de uma rota é um inteiro no intervalo $[0, \lambda]$. Portanto, considere um vértice v e um número $\varphi \in \{1, \dots, \lambda\}$. Seja $f_v(\varphi)$ a rota mais rápida de s a v com insegurança máxima φ , então $f_v(\varphi)$ satisfaz a seguinte recorrência:

$$f_v(\varphi) = \begin{cases} 0 & \text{se } \varphi = 0 \text{ e } v = s, \\ \infty & \text{se } \varphi = 0 \text{ e } v \neq s, \\ \min \left\{ \begin{array}{l} f_v(\varphi - 1) \\ \min_{u: \mu_{uv} \leq \varphi} \{ f_u(\varphi - \mu_{uv}) + \tau_{uv} \} \end{array} \right. & \text{Caso contrário} \end{cases}, \quad (5)$$

Tal recorrência naturalmente leva a um algoritmo recursivo de complexidade de tempo exponencial, porém podendo ser resolvido utilizando um algoritmo de programação dinâmica (PD) proposto por [Joksch 1966], que possui complexidade $O(|E|\lambda)$.

O Algoritmo 1 apresenta como é feito o roteamento distribuído e o encaminhamento do conhecimento para os veículos próximos ao congestionamento detectado. Primeiramente, o

Algorithm 1: Quando um veículo recebe atualização dos conhecimentos

```
Input :  $D$  // cenário com os valores  $\tau_{uv}$  e  $\mu_{uv}$  atualizados  $\forall uv \in E$   
          $p$  // porcentagem desejável de redução na insegurança da rota  
1  $D.atualizaConhecimento()$ ;  
   // Calcula atraso para reencaminhar a mensagem  
2  $T \leftarrow calculaAtraso()$ ;  
3  $s \leftarrow$  posição atual;  
4  $t \leftarrow$  destino;  
5 if  $s \neq t$  then  
   // Calcula insegurança da rota atual  
6    $\mu_P \leftarrow \sum_{uv \in P} \mu_{uv}$ ;  
7    $\lambda \leftarrow (1 - p) \cdot \mu_P$ ;  
8    $PD, \psi \leftarrow route(s, t, \lambda)$ ;  
9    $rota\_viavel \leftarrow PD[t, \lambda]$ ;  
   // Verifica se a rota é viável  
10  if  $rota\_viavel \neq \infty$  then  
    // Constrói a rota baseado em  $PD[t, \lambda]$  e seus predecessores  $\psi$   
11     $nova\ rota \leftarrow constróiRota(PD, \psi, s, t, \lambda)$ ;  
12    Altera rota;  
13  end  
14 end  
   // Aguarda  $T$  segundos  
15  $wait(T)$ ;  
16 if não ouviu a retransmissão da mensagem em  $T$  segundos then  
17   encaminhaAtualizaçãoConhecimento();  
18 end  
19 else  
20   Cancela a transmissão da atualização de conhecimento;  
21 end
```

veículo atualiza seu conhecimento baseado na mensagem recebida e calcula o atraso para o reencaminhamento da mensagem (Linhas 1 e 2). Para o roteamento, #PAS usa o algoritmo de programação dinâmica para calcular a rota mais rápida para o veículo, respeitando as restrições de segurança. Como requisito do algoritmo de programação dinâmica, o veículo precisa definir a redução desejada (p) na insegurança de sua rota, então, o veículo calcula a insegurança da sua rota atual μ_P e define $\lambda = (1 - p) \cdot \mu_P$ (Linhas 6 e 7). Portanto, baseado no valor de λ o veículo tenta otimizar a mobilidade e segurança de sua rota (Linha 8) e, sempre que uma rota é viável (i.e., o veículo pode chegar a t a partir de s com insegurança máxima de λ) o veículo altera sua rota. Por fim, se o veículo não ouviu o reencaminhamento da mensagem de atualização no período de T segundos, ele reencaminha a mensagem, caso contrário, ele cancela a transmissão.

O Algoritmo 2 descreve a função de roteamento, onde, PD é a tabela de programação dinâmica usada para memorizar a condição de tráfego para chegar ao vértice v a partir do vértice s com insegurança máxima de φ , para $s, v \in V$. Essa rota é armazenada na tabela ψ , onde $\psi[v, \varphi]$ é o predecessor de v na rota de s a v . Os dois primeiros laços (Linha 3 até 10) são responsáveis por preparar as tabelas PD e ψ no caso base, em seguida, o algoritmo completa a tabela PD com os valores de insegurança máxima de $\varphi \in \{1, 2, \dots, \lambda\}$: a rota mais rápida é a mesma rota com insegurança $\varphi - 1$ (Linha 14), ou pode ser alcançada por caminho de s a v não vazio que passa por um predecessor u e uma aresta uv (Linhas 15 até 22). Os possíveis predecessores são os vértices u que possuem $\mu_{uv} \leq \varphi$ e correspondem a uma rota de s a v , nesse caso com condição de tráfego $PD[u, \varphi - \mu_{uv}] + \tau_{uv}$ (Linha 19). Por fim, a melhor rota de s a t com insegurança máxima de λ pode ser acessada por $PD[t, \lambda]$. A rota final pode ser intuitivamente reconstruída seguindo a lista de predecessores armazenados.

Por questões de clareza, a Figura 2 mostra um digrafo representando o cenário modelado pelo sistema. A Tabela 3 exibe os valores obtidos para a tabela PD quando o algoritmo é executado no cenário da Figura 2, para um veículo com origem s e destino t , com insegurança máxima desejável de $\lambda = 10$. É importante salientar que a melhor rota para o cenário é $PD[t, 10]$.

Algorithm 2: Roteamento baseado na abordagem de programação dinâmica.

```

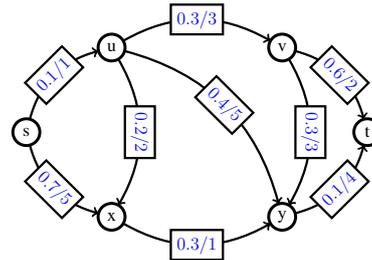
Input :  $s$  // posição atual do veículo  $n$ 
          $t$  // destino do veículo  $n$ 
          $\lambda$  // insegurança máxima desejada
Output: Rota mais rápida e segura viável de  $s$  a  $t$ .

1  $PD \leftarrow []$ ; // tabela para armazenar  $f_v(\varphi)$ 
2  $\psi \leftarrow []$ ; // armazena vértices predecessores na rota  $PD[v, \varphi]$ 
3 foreach  $\varphi \in \{1, 2, \dots, \lambda\}$  do
4    $\psi[s, \varphi] \leftarrow \emptyset$ ;
5    $PD[s, \varphi] \leftarrow 0$ ;
6 end
7 foreach  $v \in V \setminus \{s\}$  do
8    $\psi[v, 0] \leftarrow \emptyset$ ;
9    $PD[v, 0] \leftarrow \infty$ ;
10 end
11 foreach  $\varphi \in \{1, 2, \dots, \lambda\}$  do
12   foreach  $v \in V$  do
13      $\psi[v, \varphi] \leftarrow \psi[v, \varphi - 1]$ ;
14      $PD[v, \varphi] \leftarrow PD[v, \varphi - 1]$ ;
15     foreach  $uv \in E$  do
16       if  $\mu_{uv} \leq \varphi$  then
17         if  $PD[v, \varphi] > PD[u, \varphi - \mu_{uv}] + \tau_{uv}$  then
18            $\psi[v, \varphi] \leftarrow u$ ;
19            $PD[v, \varphi] \leftarrow PD[u, \varphi - \mu_{uv}] + \tau_{uv}$ ;
20         end
21       end
22     end
23   end
24 end
25 return  $PD, \psi$ ;

```

Tabela 3. Tabela PD

$v \setminus \varphi$	0	1	2	3	4	5	6	7	8	9	10
s	0	0	0	0	0	0	0	0	0	0	0
u	∞	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
x	∞	∞	∞	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3
v	∞	∞	∞	∞	0.4	0.4	0.4	0.4	0.4	0.4	0.4
y	∞	∞	∞	∞	0.6	0.6	0.5	0.5	0.5	0.5	0.5
t	∞	∞	∞	∞	∞	∞	1.0	1.0	0.7	0.7	0.6


Figura 2. Digrafo construído pelo #PAS

4. Análise dos resultados

Esta seção descreve o desempenho do #PAS. Primeiramente, a Subseção 4.1 introduz a plataforma de simulação, apresentando as ferramentas, o cenário e as métricas analisadas. A Subseção 4.2 descreve a base de dados criminais utilizada para criar o conhecimento sobre pontos de insegurança. A Subseção 4.3 avalia a efetividade e eficiência do algoritmo de roteamento distribuído em relação ao centralizado. A Subseção 4.4 compara abordagens diferentes para recomendação de rota usados em [Doolan and Muntean 2017, Ballesteros et al. 2012, Galbrun et al. 2016, Shah et al. 2011] e a Subseção 4.5 analisa o parâmetro λ usado pelo #PAS.

4.1. Metodologia

A plataforma de simulação é composta pelo simulador de mobilidade urbana (SUMO² — *Simulation of Urban Mobility*), versão 0.30.0, o simulador de rede OMNeT++³ versão 5.0 e para rede veicular foi utilizado o Veins⁴ versão 4.6. O cenário é composto por um fragmento de 5 km² do centro da cidade de São Paulo, obtido pelo OpenStreetMap⁵. Para produzir uma

²<http://sumo.dlr.de/>

³<http://omnetpp.org/>

⁴<http://veins.car2x.org>

⁵<https://www.openstreetmap.com>

mobilidade mais realista e simular um gerenciamento de tráfego próximo ao de um VNS, foram utilizadas informações sobre rotas frequentes sugeridas por VNSs nessa região^{6,7}. Além disso, para cumprir os requisitos de um VNS, as rotas mais rápidas possíveis são recomendadas para cada veículo que entra na simulação a partir as condições de tráfego atuais do cenário. Então, usando essas informações um *trace* de mobilidade de aproximadamente 2 horas foi gerado. A Tabela 4 apresenta os parâmetros adicionais utilizados na simulação. Por fim, um exemplo dos conhecimentos criados pelo #PAS são apresentados na Figura 3, onde a figura do lado esquerdo representa o conhecimento sobre condições de tráfego enquanto a figura do lado direito representa o conhecimento sobre áreas inseguras.

Tabela 4. Parâmetros

Parâmetros	Valores
Frequência do canal	5.890e9 Hz
Modelo de propagação	<i>Two ray</i>
Potência de transmissão	2.2 mW
Raio de comunicação	300 m
Bit rate	18 Mbit/s
Número máximo de saltos	10
# veículos	≈ 10.000
# Sub regiões	18
Intervalo de roteamento	450 segundos
redução desejada (p)	0.2, 0.4, 0.6, 0.8

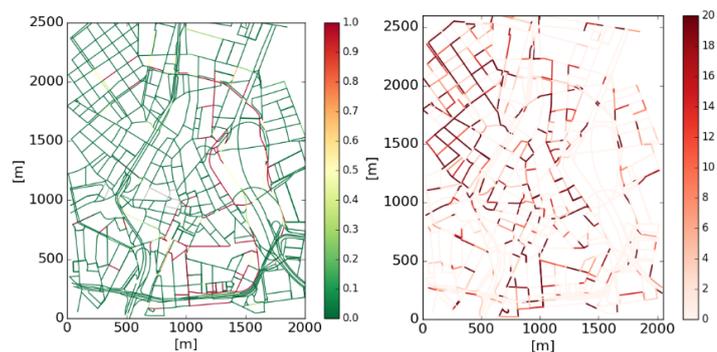


Figura 3. Conhecimentos produzidos.

4.2. Base de dados criminais

O conhecimento sobre áreas inseguras foi construído a partir das bases de dados criminais oficiais de diversos departamentos da cidade de São Paulo. A base é composta por vários boletins de ocorrência entre o período de 2002 a 2016. Entretanto, para este trabalho foram utilizados apenas os boletins relacionados a crimes de roubo, furto, assalto e sequestro dos últimos três anos, 2014 a 2016. A definição da janela temporal (2014 – 2016), foi definida para minimizar problemas de dados com qualidade ilimitada como por exemplo informações desatualizadas.

4.3. Desempenho do mecanismo para reportar informações e do algoritmo de roteamento distribuído

Para demonstrar a efetividade e eficiência do #PAS, avaliamos seu mecanismo para reportar informações dos veículos e de seu algoritmo de roteamento distribuído. Para o estudo, foi implementada uma arquitetura centralizada, na qual todos os veículos reportam suas informações para o servidor e o servidor é responsável por calcular uma nova rota para todos os veículos. Dessa forma, as seguintes métricas foram avaliadas: (i) total de mensagens transmitidas entre veículos e servidor; (ii) acurácia do conhecimento sobre as condições de tráfego criado pelo servidor; e (iii) melhoria do tempo médio de viagem em relação ao VNS.

A Figura 4 apresenta os resultados das métricas avaliadas comparando #PAS com um ITS que implementa os mesmos algoritmos de roteamento, porém, com uma arquitetura centralizada. Como pode ser observado, na arquitetura centralizada todos os veículos enviam suas informações para o servidor periodicamente, consequentemente produzindo um grande número de mensagens transmitidas, diferentemente do #PAS, o qual emprega um mecanismo

⁶<https://www.waze.com/livemap>

⁷<https://www.google.com.br/maps>

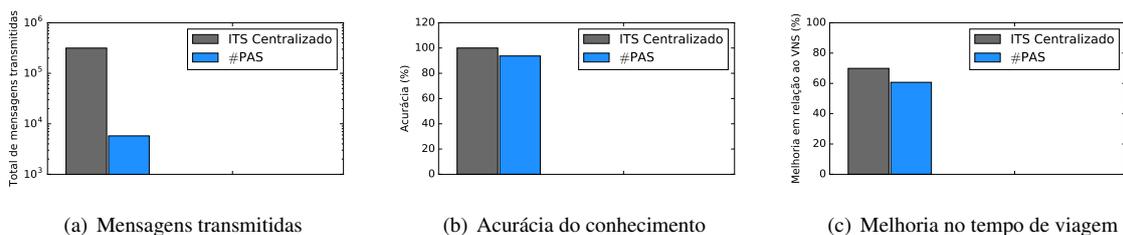


Figura 4. Desempenho do mecanismo para reportar informações e do algoritmo de roteamento distribuído

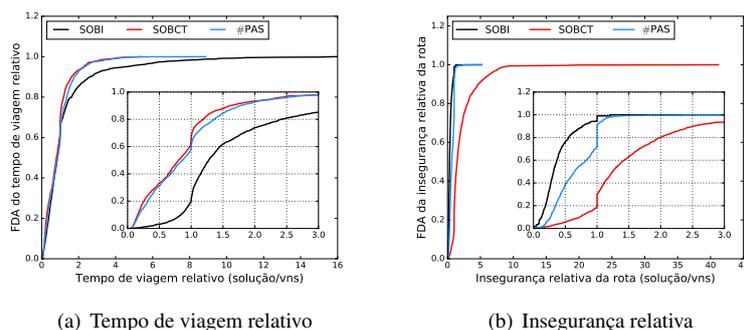


Figura 5. Resultado da comparação entre #PAS com $p=0.6$, SOBI, SOBCT

eficiente para reportar as informações de tráfego, com isso apresentando uma redução de aproximadamente 98% na quantidade de mensagens transmitidas (Figura 4(a)). Entretanto, por receber informações de todos os veículos, a arquitetura centralizada produz uma acurácia do conhecimento sobre as condições de tráfego de aproximadamente 100%, enquanto o #PAS apresenta uma acurácia de 92% devido a produção do conhecimento baseada em estimativas de conhecimento dos veículos nas sub-regiões (Figura 4(b)). Portanto, com um melhor conhecimento sobre as condições de tráfego, a arquitetura centralizada apresenta uma melhor gerência do tráfego, contudo o #PAS reduz em apenas 5% a melhoria na gerência do tráfego (Figura 4(c)).

De modo geral, o #PAS apresenta uma melhor utilização da rede com uma gerência do tráfego próxima à solução centralizada (o que é desejável). Além disso, o #PAS permite o roteamento de veículos em tempo real, uma vez que cada veículo calcula sua própria rota, enquanto na solução centralizada o servidor deve calcular a nova rota para todos os veículos.

4.4. Comparando #PAS com soluções de mobilidade e de segurança

Por questões de comparação, variamos os mecanismos de recomendação de rota para obter uma comparação justa com os mecanismos comumente utilizados pelas soluções da literatura, denominando-as como: SOBCT — *Solução Ótima Baseada nas Condições de Tráfego*, onde consideramos a recomendação de rota utilizando caminhos mínimos apenas considerando as condições de tráfego, assim como as soluções [Doolan and Muntean 2017, de Souza et al. 2017a]; e SOBI — *Solução Ótima Baseada na Insegurança*, onde consideramos a recomendação de rotas utilizando caminhos mínimos baseado nos níveis de insegurança das vias, como apresentado nas soluções [Shah et al. 2011, Galbrun et al. 2016, Santos et al. 2017]. Sendo assim, para a avaliação das soluções, foram utilizadas duas métricas, sendo elas:

- **Tempo de viagem relativo:** razão entre tempo de viagem do veículo utilizando uma das soluções (#PAS, SOBI, SOBCT) e utilizando VNS. Essa métrica resume a redução no tempo de viagem para cada veículo;

- **Insegurança relativa:** razão entre a insegurança total do veículo utilizando uma das soluções (#PAS, SOBI, SOBCT) e utilizando VNS. Essa métrica resume a redução da insegurança na rota percorrida por cada veículo.

Apesar dos VNSs serem utilizados como soluções para o problema de mobilidade urbana, ainda são pouco eficientes. Primeiramente devido à falta de um planejamento de rota dinâmico (i.e., roteamento periódico), conseqüentemente, não indicando as melhores rotas para um dado momento. Além disso, a falta de conhecimento sobre áreas inseguras pode gerar muitos problemas para motoristas e passageiros, ao serem guiados por ruas e bairros perigosos. Portanto, para mensurarmos a eficiência do #PAS e das soluções SOBI, SOBCT utilizamos os resultados do VNS implementado como *baseline*.

A Figura 5 apresenta os resultados para as métricas avaliadas, representados por um função de distribuição acumulada (FDA). Como pode ser observado, as soluções baseadas apenas nas condições de tráfego (representadas por SOBCT) visam reduzir o tempo de viagem para a grande maioria dos veículos, uma vez que reduz o tempo de viagem de aproximadamente 75% dos veículos (veja Figura 5(a)), porém, por não levarem em consideração os níveis de insegurança nas vias, essas soluções aumentam a insegurança total da rota (i.e., direcionam veículos por ruas e bairros perigosos) de mais de 70% veículos, quando comparados com um VNS (Figura 5(b)). Em particular, para alguns veículos os níveis de insegurança podem ser aumentados em até 45 vezes.

Por outro lado, as soluções que consideram apenas os níveis de insegurança nas vias (representadas por SOBI) são capazes de reduzir a insegurança nas rotas de aproximadamente 100% dos veículos em relação a um VNS. Entretanto, quando são analisadas as condições de tráfego produzidas pela solução, notamos que a mobilidade urbana piora consideravelmente, uma vez que SOBI aumenta o tempo de viagem de mais de 80% dos veículos, quando comparados com VNS. Como resultado, gerando congestionamentos em outras áreas.

A eficiência do #PAS é notável quando analisamos ambos os resultados (Figuras 5(a) e 5(b)). O #PAS, também melhora as condições de tráfego, reduzindo o tempo de viagem de aproximadamente 70% dos veículos e, apesar de aumentar o tempo de viagem para alguns veículos, esse aumento é para menos de 20% dos veículos quando comparado com VNS. Ao melhorar a mobilidade de mais de 70% dos veículos, o #PAS ainda é capaz de reduzir os níveis de insegurança na rota de aproximadamente 90% dos veículos quando comparados com VNS. É importante salientar que os resultados apresentados pelo #PAS são com o parâmetro $p = 0.6$, assim, uma redução na insegurança pode ser alcançada com valores maiores de p , entretanto, essa redução tem impacto na mobilidade dos veículos, uma descrição mais detalhada do parâmetro p é apresentado na próxima subseção.

O bom desempenho do #PAS em lidar com a mobilidade urbana e ainda aumentar a segurança dos motoristas e passageiros é consequência dos conhecimentos criados sobre condições de tráfego e níveis de insegurança. Portanto, tendo ambos os conhecimentos, o #PAS reduz os problemas de direcionar veículos por áreas inseguras para melhorar a mobilidade e também problemas de direcionar veículos para congestionamentos a fim de recomendar uma rota mais segura, quando comparados com as soluções que utilizam apenas um conhecimento para a recomendação de rotas como as soluções [de Souza et al. 2016b, de Souza et al. 2016a, Doolan and Muntean 2017, Shah et al. 2011, Galbrun et al. 2016, Ballesteros et al. 2012].

4.5. Análise exploratória do parâmetro p

A Figura 6 mostra o impacto do parâmetro p no tempo de viagem e na insegurança da rota P dos veículos, lembrando que temos $\lambda = (1 - p) \cdot \mu_P$. Os seguintes valores foram utilizados $p \in \{0.2, 0.4, 0.6, 0.8\}$.

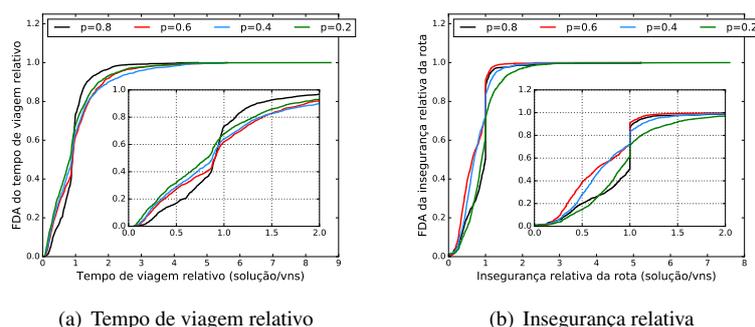


Figura 6. Resultados análise exploratória do parâmetro p .

Como pode ser visto, quanto menor o valor de p maior é a redução no tempo de viagem dos veículos (Figura 6(a)). Esse comportamento deve-se ao fato de existirem mais possibilidades para a construção da rota do veículo, uma vez que a redução na insegurança é baixa (i.e., no máximo 20% da insegurança atual para $p = 0.8$). Além disso, dependendo das características do cenário nenhuma rota alternativa pode ser construída para melhorar a mobilidade e também a insegurança. Neste contexto, para valores elevados de p muitos veículos potencialmente não irão alterar sua rota (Figura 6(b) porcentagem de veículos com valor igual a 1), e conseqüentemente, o sistema não irá melhorar a mobilidade de forma efetiva. Entretanto, apesar dos valores mais baixos de p permitirem uma melhor mobilidade, eles apresentam uma menor redução na insegurança (Figura 6(b)).

No cenário simulado, a melhor relação entre redução nas condições de tráfego e insegurança é $p = 0.6$, uma vez que com esse parâmetro #PAS reduz o tempo de viagem de aproximadamente 70% dos veículos e ainda melhora a segurança de aproximadamente 90% dos veículos.

De forma geral, apesar do impacto do parâmetro p no tempo de viagem e na redução da insegurança na rota dos veículos, #PAS apresenta uma boa relação entre melhorar a mobilidade e a segurança, pois para os valores de p utilizados sistema apresentam bons resultados, uma vez que ele é capaz de reduzir o tempo de viagem de pelo menos 60% e ainda melhorar a segurança de pelo menos 70% dos veículos

5. Conclusão

Nesse trabalho introduzimos o #PAS, um ITS híbrido para melhorar a mobilidade urbana e também a segurança dos motoristas e passageiros. Para isso, #PAS constrói um conhecimento preciso sobre as condições de tráfego e também sobre os níveis de insegurança nas vias, assim, sendo capaz de calcular rotas mais rápidas e ainda aumentar a segurança. Os resultados das simulações em um cenário realista mostraram que, quando comparado com soluções para tratar problemas de mobilidade e de segurança, #PAS apresenta um bom relação entre mobilidade e segurança. Em particular, #PAS é capaz de reduzir o tempo de viagem de aproximadamente 70% e ainda aumentar a segurança de 90%, quando comparado com um VNS tradicional. Como trabalhos futuros, pretendemos investigar alternativas para definir os níveis de insegurança nas vias e também propor um mecanismo de roteamento mais robusto, pois o algoritmo proposto é determinístico e, em cenários muito densos, onde muitos veículos possuem rotas similares, o sistema proposto pode gerar gargalos na infraestrutura de transporte, conseqüentemente degradando as condições de tráfego.

Referências

- Ballesteros, J., Rahman, M., Carbanar, B., and Rische, N. (2012). Safe cities: a participatory sensing approach. In *37th Annual IEEE Conference on Local Computer Networks*, pages 626–634.
- Board, T. R. (2010). *HCM 2010 - Highway capacity manual*. National Research Council.
- Botega, L., Oliveira, A. C. M., Perira, V. A., Saran, J. F., Villas, L. A., and Araujo, R. B. (2017). Quality-aware human-driven information fusion model. In *20th International Conference on Information Fusion*.
- (CNN), C. N. N. (2015). Waze app directions take woman to wrong brazil address, where she is killed.
- de Souza, A. M., Botega, L. C., and Villas, L. A. (2017a). Gte: Um sistema para gerenciamento de trânsito escalável baseado em compartilhamento oportunista. In *Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos - (SBRC)*.
- de Souza, A. M., Brennand, C. A., Yokoyama, R. S., Donato, E. A., Madeira, E. R., and Villas, L. A. (2017b). Traffic management systems: A classification, review, challenges, and future perspectives. *International Journal of Distributed Sensor Networks*, 13(4):1550147716683612.
- de Souza, A. M. and Villas, L. A. (2016). A fully-distributed traffic management system to improve the overall traffic efficiency. In *Proceedings of the 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, MSWiM '16*, pages 19–26, New York, NY, USA. ACM.
- de Souza, A. M., Yokoyama, R., Boukerche, A., Maia, G., Cerqueira, E., Loureiro, A. A., and Villas, L. A. (2016a). Icarus: Improvement of traffic condition through an alerting and re-routing system. *Computer Networks*, 110:118 – 132.
- de Souza, A. M., Yokoyama, R. S., Maia, G., Loureiro, A., and Villas, L. (2016b). 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.
- Djahel, S., Doolan, R., Muntean, G.-M., and Murphy, J. (2015). A Communications-Oriented Perspective on Traffic Management Systems for Smart Cities: Challenges and Innovative Approaches. *IEEE Communications Surveys Tutorials*, 17(1):125–151.
- Doolan, R. and Muntean, G. M. (2017). Ecotrec: A novel vanet-based approach to reducing vehicle emissions. *IEEE Transactions on Intelligent Transportation Systems*, 18(3):608–620.
- Galbrun, E., Pelechrinis, K., and Terzi, E. (2016). Urban navigation beyond shortest route. *Inf. Syst.*, 57(C):160–171.
- Garip, M. T., Gursoy, M. E., Reiher, P., and Gerla, M. (2015). Scalable reactive vehicle-to-vehicle congestion avoidance mechanism. In *IEEE Consumer Communications and Networking Conference (CCNC '15)*, pages 943–948.
- Jaffe, J. M. (1984). Algorithms for finding paths with multiple constraints. *Networks*, 14(1):95–116.
- Joksch, H. (1966). The shortest route problem with constraints. *Journal of Mathematical Analysis and Applications*, 14(2):191 – 197.
- Online, M. (2016). Dash cam captures the terrifying moment waze smartphone app directs a driver into a gunfight in boston.
- Pan, J., Popa, I. S., and Borcea, C. (2016). Divert: A distributed vehicular traffic re-routing system for congestion avoidance. *IEEE Transactions on Mobile Computing*, PP(99):1–1.
- Santos, F. A., Rodrigues, D. O., Silva, T. H., Loureiro, A. A., and Villas, L. A. (2017). Rotas veiculares cientes de contexto: Arcabouço e análise usando dados oficiais e sensoriados por usuários sobre crimes. In *Workshop de Gerência e Operação de Redes e Serviços - (WGRS)*.
- Shah, S., Bao, F., Lu, C.-T., and Chen, I.-R. (2011). Crowdsafe: Crowd sourcing of crime incidents and safe routing on mobile devices. In *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, GIS '11*, pages 521–524, New York, NY, USA. ACM.
- Wang, M., Shan, H., Lu, R., Zhang, R., Shen, X., and Bai, F. (2015). Real-time path planning based on hybrid-vanet-enhanced transportation system. *Vehicular Technology, IEEE Transactions on*.
- Wang, S., Djahel, S., Zhang, Z., and McManis, J. (2016). Next Road Rerouting: A Multiagent System for Mitigating Unexpected Urban Traffic Congestion. *IEEE Transactions on Intelligent Transportation Systems*, 17(10):2888–2899.