

Simulação e Análise de Congestionamento em Redes Veiculares

Mariana Ramos de Brito¹, Anna Izabel J. Tostes¹, Fátima de L.P. Duarte-Figueiredo¹

¹Departamento de Ciência da Computação

Pontifícia Universidade Católica de Minas Gerais - Belo Horizonte, MG – Brasil

rbt.mariana@gmail.com, annatostes@gmail.com, fatimafig@pucminas.br

Abstract. *Vehicular congestion is a problem in urban centers. Vehicular network management allows through communication between vehicles and infrastructure the congestion detection and the information dissemination to help drivers changing their route to a less congested one. This paper describes a congestion detection method and compares it with one from the literature. Simulations with real traffic data from Belo Horizonte were performed. The results show that the method described in this paper presents reductions of up to 17% on the mean time for the vehicles to detect congestion.*

Resumo. *O congestionamento de veículos é um problema dos centros urbanos. O gerenciamento de redes veiculares permite, através da comunicação entre veículos e de veículos com a infraestrutura, a detecção de congestionamento e a disseminação dessa informação para ajudar os motoristas a trocarem seu caminho para uma rota menos congestionada. Este trabalho descreve um método de detecção de congestionamento e compara com um método da literatura. Simulações com dados reais do trânsito de Belo Horizonte foram realizadas. Os resultados demonstram que o método descrito neste trabalho apresenta reduções de até 17% no tempo médio para os veículos detectarem o congestionamento.*

1. Introdução

O interesse na área de redes veiculares vem aumentando com o passar do tempo. Redes veiculares são formadas por veículos e equipamentos físicos, geralmente localizados às margens das vias de trânsito. Algumas das aplicações para as redes veiculares incluem a monitoração cooperativa do tráfego, o acesso à Internet e a detecção de congestionamentos. O congestionamento de veículos é um grande problema dos centros urbanos hoje em dia. As pessoas perdem muito tempo presas em congestionamentos, uma grande quantidade de combustível é gasta e a poluição ambiental sofre um aumento por causa dos agentes poluentes liberados pelos veículos [Bauza and Gozalvez 2012]. Descobrir um congestionamento e disseminar essa informação pode ajudar os motoristas a se decidirem por rotas mais vazias, reduzindo congestionamentos existentes. Nesse contexto, o problema deste artigo é como detectar um congestionamento através da rede veicular e disseminar essa informação para os outros veículos.

Este artigo descreve a simulação de uma rede que segue dados reais do trânsito de Belo Horizonte e a comparação de dois métodos de detecção de congestionamento através da rede veicular. O primeiro método abordado é da literatura recente [Fahmy and Ranasinghe 2008], chamado de agora em diante de Método da Árvore ou

de método original, e o segundo foi proposto pela aluna de Iniciação Científica (IC) Mariana Ramos de Brito [Brito et al. 2014], chamado daqui pra frente de Método da Árvore Melhorado ou método proposto. A aluna de IC propôs mudanças no método da literatura, implementou os dois métodos, simulou e comparou os resultados. O Método da Árvore utiliza uma árvore para contar o número de veículos em um congestionamento. Cada veículo troca mensagens (*beacons*) com todos os vizinhos para descobrir se está ou não em congestionamento. Entretanto, esse método apresenta algumas desvantagens, sendo elas: precisar da confirmação de todos os vizinhos na área para que um novo veículo passe a indicar congestionamento; continuar executando o algoritmo para descoberta do congestionamento quando os veículos já sabem que se encontram em um; e não levar em consideração se um veículo indica congestionamento antes de inseri-lo na árvore. O Método da Árvore Melhorado precisa apenas da maioria dos veículos para detectar o congestionamento, o algoritmo para descoberta do congestionamento só é executado enquanto o veículo sabe que não está em um e novos veículos só são anexados à árvore se indicam que estão em um congestionamento.

Os resultados foram comparados em termos de quanto tempo o último veículo que chegou à área do congestionamento levou para começar a indicá-lo; o tempo médio que os veículos, em geral, levaram para detectar o congestionamento; o tempo que levou para calcular a quantidade de veículos no congestionamento; e o número de chamadas realizadas ao algoritmo para detecção do congestionamento. O método proposto se mostrou mais eficiente que o método original. Foram obtidas reduções de 25% no tempo que o último veículo descobriu o congestionamento; 17% no tempo médio para os veículos em geral perceberem o congestionamento; 82% no tempo que a raiz da árvore levou para somar o total de veículos no congestionamento; e até 4% menos chamadas ao algoritmo de verificar a existência de congestionamento foram feitas por cada veículo da rede.

Este artigo está dividido em cinco seções. Na Seção 2 está uma breve descrição de redes veiculares e os trabalhos relacionados. Os métodos original e proposto são descritos na Seção 3. A Seção 4 traz detalhes das simulações realizadas e os resultados obtidos. Na Seção 5 estão as conclusões e trabalhos futuros.

2. Redes Veiculares e Detecção de Congestionamento

Esta seção contém uma breve descrição das redes veiculares, assim como outros trabalhos que tratam sobre detecção de congestionamento utilizando a rede veicular.

2.1. Redes Veiculares

O interesse na área de redes veiculares nos últimos anos tem aumentado consideravelmente. Redes veiculares, também conhecidas como VANETs (*Vehicle Ad hoc NETWORKS*), são redes formadas por veículos automotores e unidades fixas às margens das vias, havendo comunicação tanto entre os veículos como entre os veículos e a infraestrutura, de forma *wireless*.

VANETs são divididas em três arquiteturas atualmente [Alves et al. 2008], de acordo com a forma como os nós se organizam e se comunicam. Essas arquiteturas são: (i) *ad hoc* puro (V2V - *Vehicle-to-Vehicle*), os veículos comunicam-se apenas entre si, não existindo um nó centralizador ou elementos externos auxiliares; (ii) infraestruturada (V2I - *Vehicle-to-Infrastructure*), na qual são usados nós fixos às margens das vias, servindo

como pontos de acesso *wireless* e centralizando o tráfego da rede; e (iii) híbrida, onde existe comunicação entre os veículos, assim como uma quantidade mínima de nós fixos para aumentar a conectividade da rede. As três arquiteturas estão ilustradas na Figura 1. A arquitetura usada tanto em [Fahmy and Ranasinghe 2008] como em [Brito et al. 2014] é do tipo V2V.

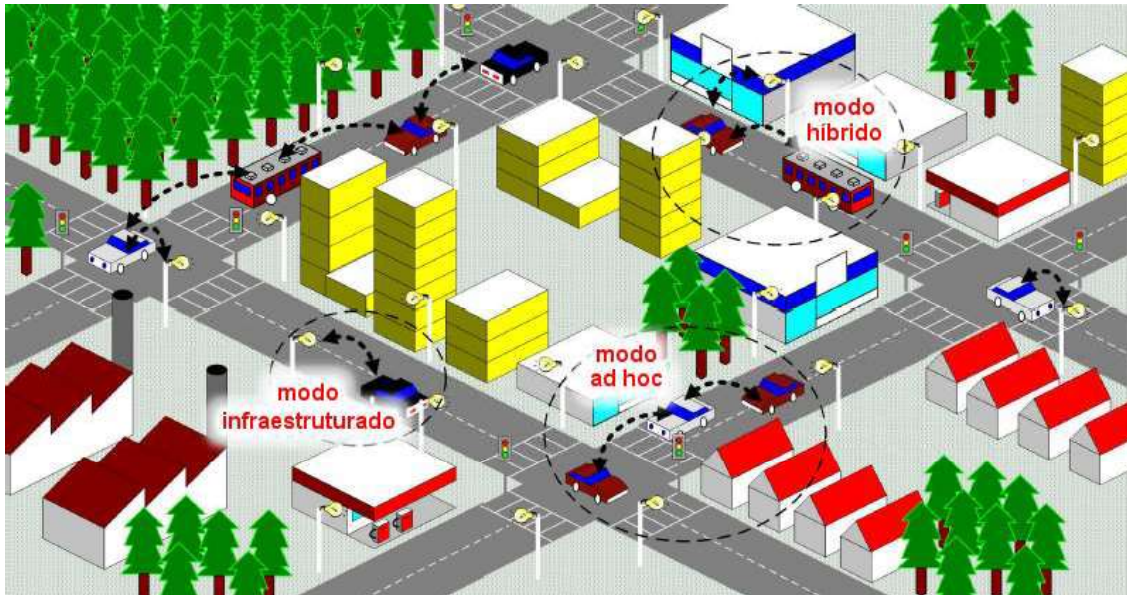


Figura 1. Arquiteturas das redes veiculares.

Fonte: Adaptado de [Alves et al. 2009].

As redes veiculares têm características únicas que as diferem de outras redes móveis *ad hoc*. A rápida velocidade com que os nós se movem e a topologia altamente mutável de uma rede veicular fazem com que os enlaces criados entre os veículos aconteçam durante apenas alguns segundos, e se quebrem quase imediatamente. Quando a densidade de veículos é baixa ou uma rota previamente estabelecida entre os nós se quebra antes de uma nova ser formada, a probabilidade de a rede ficar desconectada é maior. Isso faz com que os protocolos de roteamento usados em outras redes sem fios não sejam adequados para as VANETs [Alves et al. 2009]. Outra característica das redes veiculares é a possibilidade de modelar e prever a mobilidade dos seus nós, visto que o movimento dos mesmos é restrito pelos limites das ruas/estradas e também pelas regras de trânsito. Nas VANETs não existe preocupação com relação ao gasto de energia, visto que as baterias dos carros estão constantemente sendo recarregadas. Entretanto, atrasos na entrega dos pacotes é inviável dependendo da aplicação [Liu et al. 2009]. Se for um pacote de uma aplicação de segurança, por exemplo, o atraso na entrega de um único pacote pode fazer toda a diferença.

Existe uma lista extensa de potenciais aplicações para as redes veiculares. Algumas delas são as aplicações de segurança, que precisam divulgar informações rapidamente para que o condutor do veículo tenha tempo de reagir. Alguns exemplos que se enquadram nessa categoria são: avisos de violação dos sinais de trânsito, divulgação de informações sobre acidentes nas vias, avisos sobre condições adversas de ruas e estradas etc. Aplicações de conforto são usadas para aumentar o conforto dos passageiros e a eficiência do

tráfego. Exemplos dessas são: localização de postos de abastecimento, avisos de estacionamento, compartilhamento de músicas e filmes, troca de mensagens instantâneas etc. É preciso verificar se as exigências de uma aplicação podem ser satisfeitas, e a que nível. Um fator importante nesse sentido é a taxa de penetração necessária para o funcionamento da aplicação, ou seja, quantos veículos precisam estar equipados com a tecnologia da rede veicular [Hartenstein and Laberteaux 2008] para que a aplicação funcione.

2.2. Trabalhos Relacionados

Existem vários trabalhos que tratam da detecção de congestionamento de veículos utilizando redes veiculares. Alguns desses trabalhos, como [Wang and Tsai 2013], [Milojevic and Rakocevic 2013] e [Araújo et al. 2014] utilizam apenas comunicação veicular (V2V) para detectar o congestionamento. Já [Younes and Boukerche 2013] utiliza a forma híbrida, na qual existe comunicação entre os veículos e com a infraestrutura. Alguns métodos que utilizam apenas infraestrutura para a detecção do congestionamento são apresentados em [Bani Younes et al. 2012].

COoperative Traffic congestion detECTION - CoTEC é a técnica proposta em [Bauza and Gozalvez 2012] e se baseia em comunicação V2V e lógica *fuzzy* para detectar situações de congestionamento, sendo capaz de informar sua localização, extensão e intensidade. O CoTEC monitora as condições do tráfego como um todo através das mensagens (*beacons*) que os nós da rede (veículos) enviam randomicamente uns para os outros e, assim que uma situação de congestionamento é detectada, o CoTEC utiliza as informações de todos os veículos para caracterizar o congestionamento.

No trabalho [Terroso-Saenz et al. 2012], os autores criaram uma arquitetura movida por eventos (*Event-Driven Architecture - EDA*) que funciona com base no processamento de eventos complexos (*Complex Event Processing - CEP*). A EDA fica nos veículos da rede e também em estações centrais e fica analisando a situação do tráfego. Quando ela encontra grupos de veículos com velocidades baixas, ela infere se aquilo é um congestionamento e, se sim, informa aos veículos da rede. Entretanto, esse sistema proposto não é tão bom, pois sua eficácia depende muito da penetração da tecnologia. Os autores pretendem, futuramente, melhorar o sistema para que ele seja eficaz mesmo com uma baixa penetração.

O Método da Árvore, [Fahmy and Ranasinghe 2008], foi escolhido por se tratar de uma solução diferente das demais. A utilização de uma árvore montada com os veículos no congestionamento para contar o número de veículos é uma aplicação de uma estrutura conhecida, a árvore, para um propósito diferente.

3. Métodos Implementados

Os dois métodos foram implementados e simulados para testar os efeitos das alterações. Ambos serão descritos em mais detalhes a seguir.

3.1. Original: Método da Árvore

Os autores de [Fahmy and Ranasinghe 2008] propõem um método que monta uma árvore com os veículos envolvidos no congestionamento e consegue contá-los. Considerando que todos os veículos têm um número de identificação único (id) e que estão equipados com aparelhos *wireless*, eles enviam *beacons* (mensagens de controle) em intervalos randômicos de 3 a 6 segundos. Na Tabela 1 está a estrutura básica de um *beacon*.

Tabela 1. Estrutura de um *beacon*.

Parâmetro	Padrão	Descrição
<i>nId</i>	<i>nId</i>	Id do veículo.
<i>pId</i>	-1	Id do pai do veículo.
<i>rId</i>	<i>nId</i>	Id da raiz atualmente conhecida.
<i>HopCount</i>	1	Quantidade de saltos até a raiz.
<i>Total</i>	1	Soma dos totais dos filhos. Em nós folha é sempre 1.

Fonte: Elaborado pelos autores.

Sempre que um *beacon* é recebido, o veículo guarda o veículo emissor como vizinho e verifica se ainda é vizinho dos veículos que ele era anteriormente. Se sim, um contador de congestionamento é acrescido de 1. Quando esse contador ultrapassa o limite de 40 intervalos de tempo, ou todos os vizinhos do veículo indicam congestionamento e o contador ultrapassa 2 intervalos de tempo, o veículo muda seu estado para congestionamento e o algoritmo de montar a árvore e somar os nós começa a ser executado. 40/2 intervalos de tempo foram definidos pelos autores do trabalho original como os mais adequados para identificar o congestionamento.

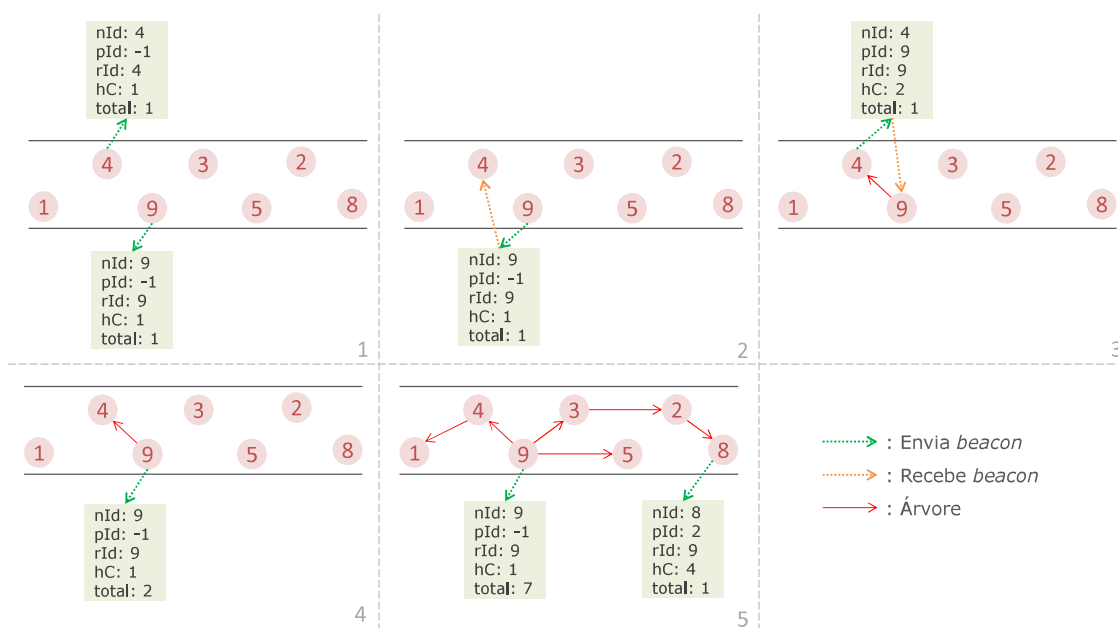


Figura 2. Montagem da árvore.

Fonte: Elaborado pelos autores.

Na Figura 2 está ilustrado o funcionamento do algoritmo de montagem da árvore. Quando o veículo recebe um *beacon* de um veículo com uma id maior que a dele para a raiz da árvore, ele muda seu parâmetro do id da raiz (rId) da árvore para o recebido e o seu parâmetro de id do pai (pId) para o id do veículo do qual ele recebeu o *beacon*. Dessa forma o veículo é anexado à árvore. Quando o veículo recebe um *beacon* vindo de um filho, ele atualiza seu parâmetro de total, somando o total recebido do filho. Assim vai até chegar na raiz da árvore, que saberá, então, o total de veículos no congestionamento.

Para impedir que ciclos sejam formados na árvore, cada veículo mantém um contador que guarda o seu número de saltos até a raiz da árvore. Se um veículo P recebe um *beacon* de um filho F que tem menos saltos que ele para a raiz da árvore, P manda uma mensagem para todos os seus filhos voltarem seus valores para o padrão e se desconectarem da árvore. Então, P também volta todos os seus valores para o padrão e se desconecta da árvore. Os veículos desconectados voltarão a fazer parte da árvore assim que receberem um *beacon* com rID maior que o conhecido.

3.2. Proposto: Método da Árvore Melhorado

Foram propostas três alterações no método original para tentar melhorar sua eficiência e o resultado dessas alterações resultou no método proposto em [Brito et al. 2014]. A montagem da árvore acontece da mesma maneira que no algoritmo original. As mudanças estão indicadas com texto sublinhado nos algoritmos abaixo.

Algorithm 1 Veículo recebeu um *beacon*

Atualiza listas de vizinhos

if Todos os vizinhos na lista de antigos estão na lista de atuais **then**

$contador \leftarrow contador + 1$

end if

if Não estou em um congestionamento^{alt 1} **then**

Chama Algoritmo 2 para verificar meu estado

end if

if Estou em congestionamento **and** Beacon recebido indica congestionamento^{alt 3} **then**

Chama algoritmo para tratar o *beacon* recebido

end if

Algorithm 2 Descobrir congestionamento

$limite1 \leftarrow 2$

$limite2 \leftarrow 40$

if Maioria^{alt 2} dos meus vizinhos em congestionamento e $cong > limite1$ **or** $contador > limite2$ **then**

Estou em um congestionamento

end if

A primeira alteração foi chamar o algoritmo de descobrir o congestionamento apenas se o veículo não está em um congestionamento. No método original, os veículos continuam executando esse algoritmo mesmo quando já sabem que estão em um congestionamento, o que leva a um processamento desnecessário. Com a alteração proposta, esse algoritmo só é executado quando necessário.

A segunda alteração proposta foi considerar as informações da maioria dos vizinhos, não de todos, para um veículo decidir seu estado. Essa alteração torna possível que os veículos se decidam pelo estado de congestionamento mais rapidamente, pois um ou outro vizinho que ainda não concorde com a situação de congestionamento não vai influenciar na sua decisão.

A terceira, e última, alteração proposta foi considerar as informações de um *beacon* recebido apenas quando seu veículo emissor também indica congestionamento. Essa alteração foi proposta para evitar o problema causado por uma “raiz falsa”. Seguindo o método original, os veículos começam a executar o algoritmo de montar a árvore quando passam a indicar congestionamento, mas não há uma preocupação se os veículos sendo anexados à árvore estão realmente no congestionamento. Se um veículo A com id mais alto do que os que estão no congestionamento passa perto da área congestionada, em uma rua paralela por exemplo, e entra em contato com algum veículo B no congestionamento, o veículo B pode achar que o veículo A é a raiz da árvore e começar a avisar aos outros veículos sobre A ser a raiz. Como A não está no congestionamento, ele logo sai da área de alcance, mas os veículos continuam achando que ele é a raiz por um tempo. A alteração proposta evita esse problema, pois apenas se um veículo indica congestionamento ele é adicionado à árvore. Portanto, a raiz falsa (veículo de id alto que não participa do congestionamento, mas que entra no alcance de algum veículo no congestionamento) nunca é considerada a raiz da árvore quando os veículos seguem o método proposto.

4. Simulações Realizadas e Resultados

Foi criada uma rede simulada que segue dados reais do trânsito do centro de Belo Horizonte. Os dados utilizados para criar o fluxo de veículos foram cedidos pela BH Trans¹, que é a empresa responsável pelos transportes e pelo trânsito de Belo Horizonte. Simulação foi a forma escolhida para testar os métodos implementados porque testes reais são caros e podem ser perigosos. Os simuladores usados para este trabalho foram o SUMO², que é o simulador de mobilidade; o OMNeT++³, simulador de rede; e o Veins⁴, que é a plataforma de integração entre os dois simuladores.

4.1. Cenários Simulados

Como o fluxo de veículos segue dados reais do trânsito, os tempos de simulação foram variados para obter tamanhos diferentes de rede. Para uma rede com 61 veículos no total e 54 no congestionamento, o tempo de simulação foi de 10 minutos. A rede de 62 veículos ao todo e 54 no congestionamento também foram 10 minutos de simulação, com a adição de um veículo especialmente modificado que será explicado. Para 86 veículos ao todo, 75 no congestionamento, as simulações foram de 15 minutos. Com 20 minutos de simulação, a rede foi composta por 111 veículos ao todo, sendo 99 no congestionamento.

O veículo especial inserido na rede de 62 veículos não participa do congestionamento, apenas passa em uma rua lateral a qual ele acontece e se conecta brevemente com os veículos no congestionamento. Esse veículo foi modificado para ter o maior número id da rede e foi inserido para testar os efeitos que uma “raiz falsa” tem nos dois métodos testados.

4.2. Resultados

O gráfico da Figura 3 exibe os instantes no tempo da simulação, em minutos, que o último veículo de cada tamanho de rede/método percebeu que estava em um congestionamento.

¹BH Trans: <http://www.bhtrans.pbh.gov.br>

²SUMO: <http://sumo-sim.org>

³OMNeT++: <http://www.omnetpp.org>

⁴Veins: <http://veins.car2x.org>

O tempo exibido no gráfico é o tempo desde que o veículo saiu da origem até chegar ao congestionamento e percebê-lo.

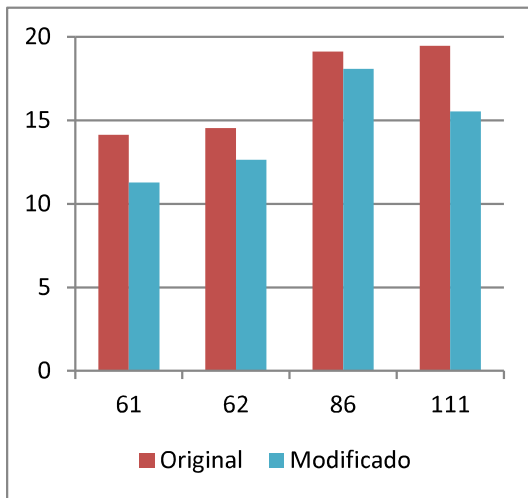


Figura 3. Último veículo.

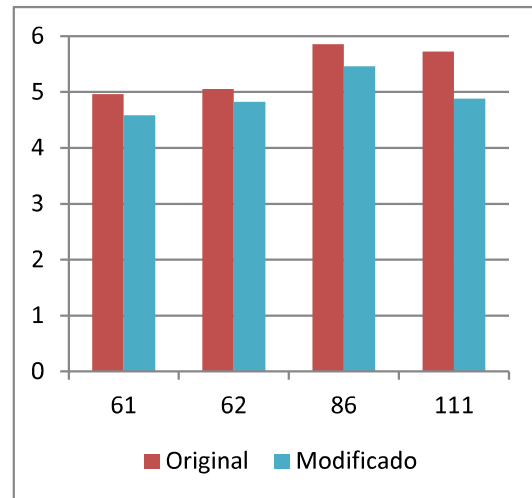


Figura 4. Tempo médio.

Pela Figura 3 é possível notar que, dos veículos que chegaram por último à área congestionada, os que seguiam o método melhorado conseguiram perceber o congestionamento até 25% mais rapidamente que os veículos seguindo o método original. Isso se deve à alteração de considerar as informações apenas da maioria dos vizinhos, ou seja, os últimos veículos conseguem perceber o congestionamento em apenas 2 intervalos de tempo, enquanto que no método original isso nem sempre acontece.

No gráfico da Figura 4 estão os tempos médios, em minutos, que os veículos precisaram, de fato, para perceber o congestionamento. Como é possível perceber, o método proposto foi novamente mais eficiente que o método original, devido à alteração de considerar as informações apenas da maioria dos vizinhos.

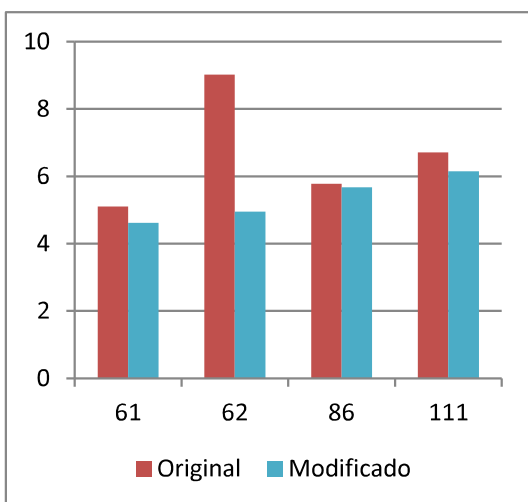


Figura 5. Raiz somou total.

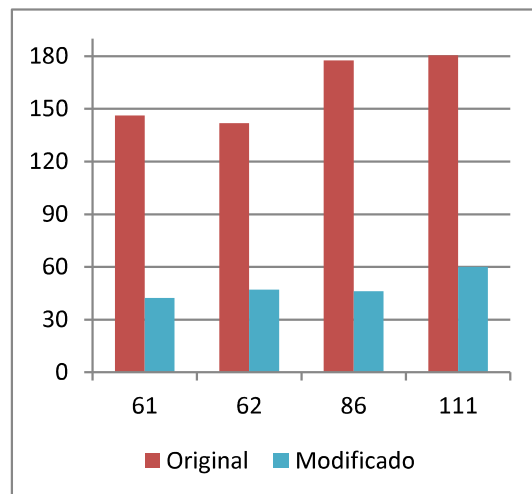


Figura 6. Chamadas ao algoritmo de descoberta.

A Figura 5 traz o gráfico com os instantes no tempo, em minutos, que a raiz

verdadeira da árvore somou o total de veículos no congestionamento. Os veículos raiz seguindo o método proposto conseguem somar o total de veículos mais rapidamente, pois os veículos em geral descobriram o congestionamento de forma mais rápida e a raiz só pode somar os nós que indicam congestionamento. No caso da rede especial, a de 62 veículos, o veículo raiz seguindo o método original demorou consideravelmente mais tempo para somar o total de veículos, pois ele perdeu tempo acreditando que a raiz falsa era a verdadeira.

No gráfico da Figura 6 estão os totais das chamadas que foram feitas ao algoritmo de descobrir o congestionamento. Devido à alteração de parar de chamar esse algoritmo quando o veículo já indica congestionamento, os veículos seguindo o método proposto realizaram muito menos chamadas. Os veículos que seguiam o método original continuaram as chamadas a esse algoritmo mesmo quando já indicavam congestionamento, o que tornou seu processamento maior.

5. Conclusões

Neste trabalho foi descrito o método de detecção de congestionamento através de redes veiculares proposto em [Brito et al. 2014]. Este método foi baseado no método proposto em [Fahmy and Ranasinghe 2008], o Método da Árvore, sendo que foram propostas alterações que visavam melhorar sua eficiência. Como foi possível perceber, este objetivo foi alcançado e o método resultante, o Método da Árvore Melhorado, obteve melhoras de 25% no tempo que o último veículo descobriu o congestionamento; 17% no tempo médio para os veículos em geral perceberem o congestionamento; 82% no tempo que a raiz da árvore levou para somar o total de veículos no congestionamento; e até 4% menos chamadas ao algoritmo de verificar a existência de congestionamento foram feitas por veículo.

Possíveis trabalhos futuros a este incluem: otimização no envio dos *beacons*, integração com técnicas de minimização de congestionamentos, testes em outros cenários e comparação com outros métodos de detecção de congestionamento.

Referências

- Alves, R. d. S., Abdesslem, F. B., Cavalcanti, S. R., Campista, M. E. M., Costa, L. H. M., Rubinstein, M. G., de Amorim, M. D., and Duarte, O. C. M. (2008). Uma análise experimental da capacidade de redes ad hoc veiculares. *SBrT - Simpósio Brasileiro de Telecomunicações*, page 8.
- Alves, R. S., Campbell, I. V., Couto, R. S., Campista, M. E. M., Moraes, I. M., Rubinstein, M. G., Costa, L. H. M. K., Duarte, O. C. M. B., and Abdalla, M. (2009). Redes veiculares: Princípios, aplicações e desafios. In *27^o Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, pages 199–254, Recife. Minicurso do Simpósio Brasileiro de Redes de Computadores.
- Araújo, G. B., Duarte-Figueiredo, F. d. L. P., Tostes, A. I. J., and Loureiro, A. A. F. (2014). Um Protocolo de Identificação e Minimização de Congestionamentos de Tráfego para Redes Veiculares. In *Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*.

- Bani Younes, M., Roman Alonso, G., and Boukerche, A. (2012). Time-distance path recommendation mechanisms for vanets. In *Proceedings of the Second ACM International Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications*, DIVANet '12, pages 75–80, New York, NY, USA. ACM.
- Bauza, R. and Gozalvez, J. (2012). Traffic congestion detection in large-scale scenarios using vehicle-to-vehicle communications. *Journal of Network and Computer Applications*, 36(5):1295 – 1307.
- Brito, M. R., Tostes, A. I. J., Duarte-Figueiredo, F. d. L. P., and Loureiro, A. A. F. (2014). Simulação e Análise de Métodos de Detecção de Congestionamento de Veículos em VANET. In *Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos. Workshop de Gerência e Operação de Redes e Serviços - WGRS*.
- Fahmy, M. F. and Ranasinghe, D. N. (2008). Discovering automobile congestion and volume using vanet's. In *ITS Telecommunications - ITST. 8th International Conference on*, pages 367 –372.
- Hartenstein, H. and Laberteaux, K. P. (2008). A tutorial survey on vehicular ad hoc networks. *IEEE Communications Magazine*, 46(6):164–171.
- Liu, Y., Bi, J., and Yang, J. (2009). Research on Vehicular Ad Hoc Networks. In *21st Chinese Control and Decision Conference, Vols 1-6, Proceedings*, pages 4430–4435. NE Univ; IEEE Ind Elect Singapore Chapter; Guilin Univ Elect Technol; IEEE Control Syst Soc; IEEE Ind Elect Soc.
- Milojevic, M. and Rakocevic, V. (2013). Short paper: Distributed vehicular traffic congestion detection algorithm for urban environments. In *Vehicular Networking Conference (VNC), IEEE*, pages 182–185.
- Terroso-Saenz, F., Valdes-Vela, M., Sotomayor-Martinez, C., Toledo-Moreo, R., and Gomez-Skarmeta, A. F. (2012). A Cooperative Approach to Traffic Congestion Detection With Complex Event Processing and VANET. *IEEE Transactions on Intelligent Transportation Systems*, 13(2):914–929.
- Wang, C. and Tsai, H.-M. (2013). Detecting urban traffic congestion with single vehicle. In *Connected Vehicles and Expo (ICCVE), International Conference on*, pages 233–240.
- Younes, M. and Boukerche, A. (2013). Efficient traffic congestion detection protocol for next generation vanets. In *Communications (ICC), IEEE International Conference on*, pages 3764–3768.