

RouteSpray: Um algoritmo de roteamento de múltiplas cópias baseado em rotas de trânsito

Maurício José da Silva¹, Fernando Augusto Teixeira², Ricardo R. Oliveira¹

¹Universidade Federal de Ouro Preto (UFOP)
Ouro Preto – MG – Brasil

²Universidade Federal de São João Del Rey (UFSJ)
Ouro Branco – MG – Brasil

badriciobq@gmail.com, teixeira@ufsj.edu.br, rrabelo@gmail.com

Abstract. *Vehicular networks are a special kind of wireless networks that have gained the attention of researchers in the last few years. Routing protocols for this kind of network have to deal with several challenges such as high mobility, high speeds and frequent network disconnections. In this paper the RouteSpray is proposed, which is an algorithm of vehicular routing that, in addition to using vehicular routes to help making routing decisions, also uses controlled spraying in order to forward multiple copies of messages, thus ensuring better delivery rates without overloading the network. The results of the experiments show that the RouteSpray delivered 13.46% more messages than other proposals in the literature and kept the buffer occupation 73.38% lower.*

Resumo. *Redes veiculares são um tipo especial de redes wireless que ganharam a atenção dos pesquisadores nos últimos anos. Protocolos de roteamento para esse tipo de rede têm que lidar com diversos desafios como alta mobilidade, altas velocidades e frequentes desconexões na rede. Neste artigo é proposto o RouteSpray, um algoritmo de roteamento veicular que, além de utilizar as rotas dos veículos para tomar as decisões de roteamento, também utiliza a pulverização controlada para encaminhar múltiplas cópias de mensagens, garantindo melhores taxas de entrega sem sobrecarregar a rede. Os resultados dos experimentos mostram que o RouteSpray entregou 13,46% mensagens a mais do que outras propostas da literatura e manteve a ocupação do buffer 73,38% menor.*

1. Introdução

Redes Veiculares *ad hoc* (VANETs) são um tipo especial de redes *wireless* que ganharam a atenção dos pesquisadores nos últimos anos. Esses tipos de rede oferecem, a partir dos Sistemas de Transporte Inteligentes (ITS), serviços como assistência ao motorista, disseminação de informações e entretenimento [Taysi and Yavuz 2012].

Em redes VANETs, a alta mobilidade dos veículos provoca frequentes desconexões entre os nós da rede. Tal característica particiona a rede e impossibilita a utilização dos protocolos de roteamento feitos para redes *ad hoc*. Porém, algumas características das redes VANETs podem ser utilizadas para auxiliar no roteamento, como por exemplo, padrões de mobilidade limitados pelas rodovias, tendência dos veículos se locomoverem

em grupos e a integração de sensores aos veículos [Toor et al. 2008] [Li and Wang 2007]. Há vários algoritmos de roteamento propostos para VANETs, mas os recentes avanços tecnológicos e sua popularização como aconteceu com o GPS, por exemplo, abriram a possibilidade de propor protocolos ainda mais eficientes.

Neste artigo é proposto o algoritmo *RouteSpray*. Tal algoritmo combina quatro conceitos importantes para tomar as decisões de roteamento, sendo: (i) utilização da técnica de *store-carry-and-forward* [Zhao and Cao 2008] para rotar as mensagens; (ii) transmissão das mensagens baseando-se em contato direto [Spyropoulos et al. 2008a]; (iii) utilização das rotas dos veículos para auxiliar no roteamento, considerando-se que os veículos são equipados com GPS e (iv) utilização da técnica de pulverização controlada de mensagens [Spyropoulos et al. 2008b].

Vários protocolos que objetivam realizar o roteamento em redes VANETs têm sido apresentados. A principal diferença entre esses protocolos são as informações que eles consideram para realizar o roteamento (histórico de contato entre os nós, informações de localização, etc) e a estratégia que utilizam para encaminhar a mensagem (número de réplicas geradas por mensagem). Contudo, existe um consenso entre a comunidade científica de que não existe um protocolo de roteamento ideal para todos os cenários. O protocolo *RouteSpray* tem o objetivo de ser aplicado em cenários onde tem-se conhecimento prévio das rotas dos veículos, característica que tornou-se comum nos dias de hoje devido à popularização dos dispositivos de navegação, principalmente se considerarmos frotas com mobilidade controlada, como ônibus, caminhões e táxis. Tal protocolo foi projetado para utilizar as rotas dos veículos para tomar as decisões de roteamento, sendo essa a única premissa exigida.

O protocolo *RouteSpray* foi validado através de uma simulação de um ambiente urbano onde os veículos percorrem rotas que ligam os pontos de interesse de uma cidade. Para garantir uma simulação mais próxima da realidade, foi utilizado o simulador OMNeT++ [Varga 1999] juntamente com o VeNeM [Silva 2012]. Os resultados obtidos mostraram que o algoritmo *RouteSpray* entregou 13,46% mensagens a mais do que outras propostas da literatura e manteve a ocupação do *buffer* 73,38% menor.

O restante do artigo está organizado da seguinte forma: na Seção 2 são apresentados os trabalhos relacionados. Na Seção 3 é apresentado o funcionamento do *RouteSpray*. Na Seção 4 são apresentados o ambiente de simulação e os resultados. Finalmente, na Seção 5 são apresentados as conclusões e os trabalhos futuros.

2. Trabalhos Relacionados

Nos últimos anos, vários trabalhos propuseram resolver o problema de roteamento em redes *ad hoc*, como o DSR (*Dynamic Source Routing*) [Johnson and Maltz 1996] e o AODV (*Ad-hoc On-demand Distance Vector*) [Perkins and Royer 1999]. Esses protocolos iniciam a transmissão de dados somente após estabelecerem um caminho entre a origem e o destino, característica que na maior parte do tempo não é satisfeita em VANETs. Isso, porque esse tipo de rede sofre frequentes desconexões provocadas pela alta velocidade e alta mobilidade dos veículos. Com a finalidade de evitar perdas de dados, os protocolos de roteamento para VANETs consideram a utilização da técnica de *store-carry-and-forward* [Lee and Gerla 2010].

Outra característica que pode beneficiar algoritmos de roteamento para VANETs,

é a utilização de informações de localização dos nós da rede, que tornou-se possível devido à integração de Sistemas de Navegação (NS) aos veículos. Protocolos de roteamento que utilizam essa técnica são classificados como protocolos *position-based*, ou *geographic-based* [Allal and Boudjit 2012].

Em redes VANETs, a transmissão de forma oportunista e informações de contexto podem ser utilizadas para melhorar o processo de roteamento. Um esquema que explora a transmissão oportunista para garantir a entrega da mensagem é chamado roteamento Epidêmico [Vahdat et al. 2000]. O algoritmo de roteamento Epidêmico utiliza a técnica de *store-carry-and-forward* para melhorar a taxa de entrega de dados. Ele armazena as mensagens recebidas em um *buffer* e aproveita-se do contato oportunista para replicar as mensagens armazenadas para os outros nós da rede. Essa técnica provoca inundação de mensagens na rede e garante que uma das réplicas da mensagem siga o menor caminho existente até o destino. Com isso, o algoritmo Epidêmico consegue a melhor taxa de entrega de mensagens e o menor atraso para que a mensagem alcance o destino. Porém, devido ao número excessivo de réplicas de mensagens, esse tipo de roteamento provoca degradação da rede, gastos abusivos de energia e grande ocupação dos *buffers*. Essas características inviabilizam a utilização do algoritmo Epidêmico em diversos cenários.

Para resolver os problemas apresentados pelo algoritmo Epidêmico, em [Spyropoulos et al. 2008b] é proposto um algoritmo chamado *Spray and Wait*. O algoritmo *Spray and Wait* utiliza a técnica de “*spray*” para diminuir o número de réplicas de mensagens pulverizadas pela rede. Apesar dos autores introduzirem sua utilização em redes esparsas, a técnica de “*spray*” foi utilizada pela primeira vez em redes de celulares. Sua utilização objetivava pulverizar as mensagens entre os pontos que os usuários mais frequentavam [Tchakountio and Ramanathan 2001]. O algoritmo *Spray and Wait* é dividido em duas fases. Na fase “*spray*”, o nó de origem calcula o número de cópias que devem ser pulverizadas. Esse cálculo é baseado no número de nós da rede e no tempo de atraso desejado para que a mensagem alcance o destino. Essas cópias são pulverizadas de forma oportunista entre os nós que entram na área de transmissão do nó de origem. Se a mensagem não for entregue ao destino na fase “*spray*”, os nós iniciam a fase “*wait*”. Na fase “*wait*”, o nó mantém a mensagem em seu *buffer* até encontrar com o nó de destino, para somente então fazer a entrega da mensagem.

Apesar do algoritmo *Spray and Wait* se beneficiar de redes densas, em redes veiculares essa característica se torna uma armadilha e compromete o desempenho do algoritmo. Isso acontece porque o fluxo de carros se concentra em cruzamentos ou em semáforos, e os veículos se dividem entre as opções de direções que podem seguir. Tal comportamento faz com que algumas cópias das mensagens sejam levadas para longe do destino. Para resolver esse problema, o algoritmo *RouteSpray* utiliza as rotas dos veículos para escolher qual é a melhor opção para carregar a mensagem. Ele pulveriza as mensagens somente entre os nós que encontrarão com o destino, e com isso, impede que sejam pulverizadas desnecessariamente para nós que nunca poderão entregá-las.

3. Algoritmo *RouteSpray*

Para realizar o roteamento, o protocolo *RouteSpray* assume que os veículos são equipados com GPS e que todas as suas rotas são conhecidas. Além disso, não é necessária a existência de nenhuma infraestrutura fixa de rede, ou seja, é possível realizar o roteamento

entre os veículos de forma totalmente *ad hoc*.

O funcionamento do protocolo é baseado no uso de dois tipos de mensagens, sendo elas as mensagens de controle e as mensagens de dados. As mensagens de controle são utilizadas para manter o estado da rede, enviando informações de contexto para os nós vizinhos. A comunicação inicia-se através de um *handshake*, onde os nós trocam informações dos pacotes que já foram entregues pela rede, permitindo fazer o controle de mensagens armazenadas em *buffer*, o que é conseguido apagando as que já foram entregues. Essa característica é desejável em redes móveis.

Outra função das mensagens de controle é a troca de informações sobre o estado do *buffer*. O nó de origem envia para seus vizinhos uma lista contendo um identificador e o destino de cada mensagem que possui em seu *buffer*. Com essas informações o vizinho calcula, utilizando as rotas pré-estabelecidas, o tempo em segundos que demorará para entregar cada mensagem. Após a origem receber a resposta do nó vizinho, ela é capaz de decidir qual é o melhor transportador para a mensagem. Todo o processo de roteamento descrito anteriormente pode ser visto com mais detalhes no Algoritmo 1.

```
Input: mensagem
if recebeu mensagem de controle then
  if mensagem de handshake then
    LimpaBuffer;
    AtualizaListaDeMensagensEntregues;
    EntregaMensagensEndereçadasOrigem;
    RespondeListaMensagensBuffer();
  else if resposta a mensagem de handshake then
    RecebeRelaçãoMensagensVizinho;
    EnviaTempoDeContatoMensagem();
  else
    ProcessaRespostaTempoDeContato;
    DecideQualMelhorTransmissor();
  end
else if recebeu mensagem de dados then
  if mensagem endereçada a mim then
    ProcessarMensagem();
  else
    ArmazenarNoBuffer();
  end
end
```

Algoritmo 1: Pseudo-código *RouteSpray*

A melhoria no desempenho do *RouteSpray* se dá pela combinação de dois conceitos importantes: (i) utilização de rotas para obter conhecimento prévio dos contatos entre os nós e (ii) utilização da técnica de *Binary Spray*. Tal combinação garante melhores taxas de entrega sem sobrecarregar a rede. Ambos os conceitos são explicados com mais detalhes a seguir.

3.1. Utilização de Rotas

Informações de geolocalização em uma rede permitem ter conhecimento prévio sobre a posição de seus nós. Tal característica possibilita o encaminhamento de pacotes na

direção do destino e melhora a taxa de entrega de dados. A utilização das rotas dos veículos garante que o algoritmo consiga prever os contatos entre os nós da rede. Assim, ele pode tomar a melhor decisão de encaminhamento.

Considerando que três veículos seguirão rotas pré-estabelecidas (Figura 1), e que o veículo *B* possui um pacote destinado ao veículo *C*, apesar das rotas dos veículos *B* e *C* se cruzarem o veículo *B* escolherá o veículo *A* como melhor transportador da mensagem até o destino. Isso porque o veículo *A* se encontrará com o veículo *C* antes do veículo *B*. Esse processo garante que a mensagem seja entregue no menor tempo possível.

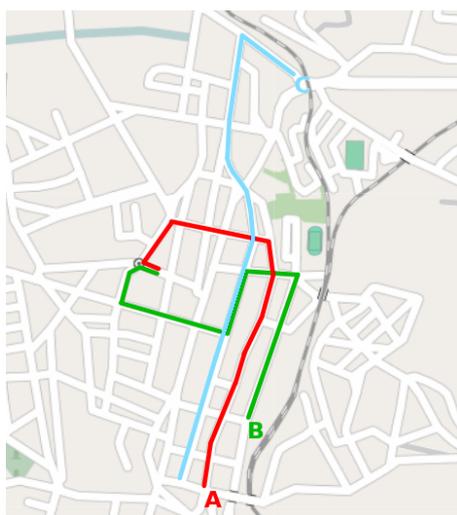


Figura 1. Rotas pré-estabelecidas para três veículos.

3.2. Binary Spray

Esquemas de encaminhamento baseados em uma única cópia de mensagem provocam grandes atrasos na entrega. Por outro lado, esquemas de encaminhamento baseados em inundação provocam degradação na rede. Visando obter o menor atraso na entrega sem degradar a rede, em [Spyropoulos et al. 2008b] os autores propuseram a técnica de “*spray*”, que consiste em gerar um número controlado de cópias de mensagens e pulverizá-las entre os nós da rede. Quando um veículo deseja transmitir uma mensagem, ele gera um número controlado de cópias (L). Para calcular o valor de L , é levado em consideração o número de nós presentes na rede e o tempo desejado de atraso para que a mensagem alcance o destino. A pulverização pode acontecer de duas formas diferentes, as quais os autores deram os nomes de *Source Spray* e *Binary Spray*.

Na pulverização baseada no *Source Spray*, o nó de origem encaminha as L cópias da mensagem para os primeiros L nós distintos que encontrar. No *Binary Spray*, o nó de origem inicia com L cópias; enquanto o nó A possui $n > 1$ cópias (for a origem ou o transportador) e encontrar com outro nó B (que não possui nenhuma cópia) ele entregará ao nó B $\lfloor (n/2) \rfloor$ cópias e manterá $\lceil (n/2) \rceil$ consigo mesmo; quando o nó possuir somente uma cópia, ele escolherá o contato direto para realizar a entrega.

Uma característica do *Source Spray* é que a mensagem realiza somente dois saltos até atingir o destino, ou seja, o origem encaminha a mensagem para o transportador tornando-o o responsável por entregá-la ao destino. Essa característica provoca atrasos

na entrega em redes com mobilidade controlada, o que inviabiliza seu uso em VANETs. Já no *Binary Spray*, o fato do nó transportador receber mais de uma cópia da mensagem encaminhando-as em contatos futuros, faz com que esta técnica realize o encaminhamento baseado em múltiplos saltos, diminuindo o tempo de entrega da mensagem. Por este motivo, optou-se por utilizar o *Binary Spray* como política de encaminhamento de mensagem no *RouteSpray*.

4. Análise de desempenho

Apesar de ser desejável a avaliação de protocolos de roteamento em ambientes reais, o alto custo de implementação e a dificuldade em mobilizar pessoal suficiente para realizar os experimentos tornam inviável tal implementação. Por esse motivo, a comunidade científica avalia os protocolos de roteamento através de simulações. Para avaliar o desempenho do *RouteSpray*, foram realizadas simulações comparando-o com os protocolos Epidêmico e *Spray and Wait*, que são os principais protocolos de roteamento existentes para redes esparsas.

Certas propriedades são desejáveis em algoritmos de roteamento, como escalabilidade, que é a capacidade do algoritmo se adaptar a redes com diferentes densidades; robustez, que é a capacidade de realizar o roteamento mesmo em caso de falhas, que podem ser provocadas em VANETs, por exemplo, por obstáculos que impedem as transmissões; simplicidade, que é a facilidade de implementação; e otimização, que pode ser aplicada a vários requisitos de roteamento (taxa de entrega de mensagens, *throughput*, etc). O desempenho do protocolo *RouteSpray* foi avaliado segundo as seguintes métricas: (i) taxa de entrega de mensagens; (ii) ocupação dos *buffers* e (iii) quantidade de mensagens enviadas na rede. A taxa de entrega de mensagens refere-se ao número de mensagens entregues ao destino, e é importante para comprovar a eficácia do protocolo. A ocupação dos *buffers* é definida como a soma de todas as mensagens armazenadas nos nós da rede, e deve ser analisada porque dispositivos que são utilizados em redes móveis possuem restrições de armazenamento. Já a quantidade de mensagens enviadas na rede é a soma de todas as mensagens enviadas, incluindo as de controle, e indica a quantidade de transmissões necessárias para garantir a entrega das mensagens.

O cenário de simulação utilizado consiste em um ambiente urbano, onde foram geradas 16 rotas entre os pontos de interesse de uma cidade. A quantidade de veículos na rede variou entre 5 e 100, gerando redes com diferentes densidades. Para simulações cujo número de veículos é maior do que o número de rotas, mais de um veículo realiza o mesmo percurso. Nesse caso, além dos veículos serem distribuídos de forma uniforme entre as rotas, eles partem em tempos diferentes. Considerou-se ainda que a velocidade de transmissão é maior do que a velocidade de locomoção.

Para a simulação, foi utilizado o framework MiXiM [Köpke et al. 2008], que é uma extensão ao simulador de redes OMNeT++ [Varga 1999]. Simulações de mobilidade veicular baseadas em modelos de mobilidade randômicos, não correspondem à realidade [Gamess et al. 2012]. Isso, porque a movimentação dos veículos esta limitada às restrições das ruas e avenidas. Além disso, parâmetros como velocidade e direção sofrem variações. Por esse motivo, a mobilidade veicular foi gerada utilizando o software VeNeM [Silva 2012].

Os parâmetros utilizados na simulação podem ser conferidos na Tabela 1.

Parâmetros	Valores
Tempo de simulação	2700s
Playground X	2.41km
Playground Y	4.05km
Número de nós	5, 10, 25, 50, 75, 100
Valores de L	2, 3, 6, 8, 12, 20
Frequência de banda	2.4GHz
Potência de transmissão	110.11mW
Atenuação de sinal	-70dBm
Tamanho do pacote	512bits

Tabela 1. Parâmetros utilizados na simulação

4.1. Resultados

Quando o cenário não impõe restrições de armazenamento, o algoritmo Epidêmico realiza a entrega de todas as mensagens enviadas, o que o torna uma importante ferramenta de comparação entre os algoritmos de roteamento. Outro algoritmo que consegue taxa de entrega semelhante ao Epidêmico e ainda provoca menor degradação da rede, é o *Spray and Wait*. Por esse motivo, o algoritmo *RouteSpray* foi comparado com ambos os algoritmos. Considerando que o algoritmo Epidêmico entrega 100% das mensagens enviadas, os algoritmos *RouteSpray* e *Spray and wait* entregaram 57,66% e 44,23%, respectivamente. Os valores obtidos são apresentados mais detalhadamente na Figura 2. A melhor taxa de entrega de mensagens é obtida pelo algoritmo *RouteSpray*, consequência da utilização das rotas dos veículos.

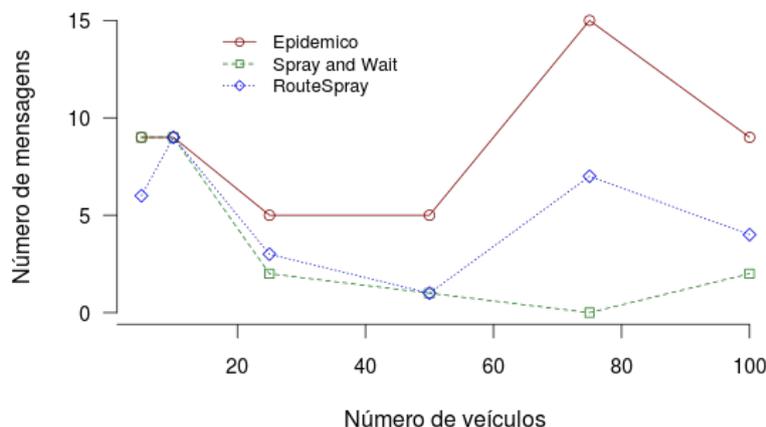
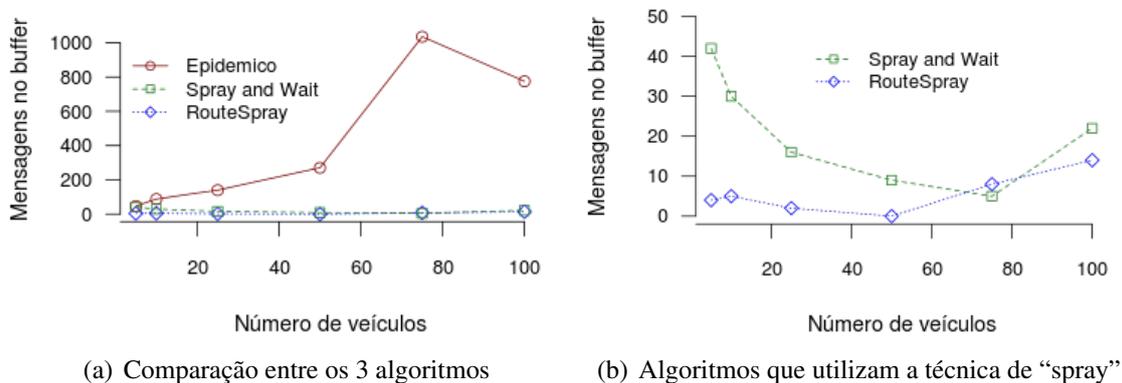


Figura 2. Mensagens entregues na rede.

Como mostrado na Figura 3(a), o algoritmo Epidêmico exige que os nós possuam grandes capacidades de armazenamento, porque cada nó da rede armazena uma cópia de cada mensagem transmitida. Essa característica pode ser percebida mais claramente observando-se as transmissões necessárias para a rede com 75 nós onde, para entregar 15 mensagens, o algoritmo Epidêmico armazenou 1035 cópias destas. Já os algoritmos *Spray and Wait* e o *RouteSpray*, provocaram pouca ocupação dos *buffers*, o que prova a eficiência da técnica de *spray*, como podemos ver na Figura 3(b). O algoritmo *RouteSpray* obteve uma ocupação em *buffer* 73,38% menor do que o *Spray and Wait*, pois faz controle das mensagens armazenadas em *buffer*, apagando as que já foram entregues.



(a) Comparação entre os 3 algoritmos (b) Algoritmos que utilizam a técnica de “spray”

Figura 3. Ocupação dos buffers

Conforme apresentado, a utilização de mensagens de controle pelo algoritmo *RouteSpray* permite melhorar as taxas de entrega de dados e controlar a ocupação do *buffer*. Porém, elas introduzem um custo adicional à rede, provocando um número maior de transmissões de mensagens. Tal característica pode ser vista na Figura 4. Apesar do algoritmo *RouteSpray* provocar mais transmissões de mensagens do que o algoritmo Epidêmico, ele provoca menor sobrecarga na rede devido ao fato de as mensagens de controle não possuírem carga de dados.

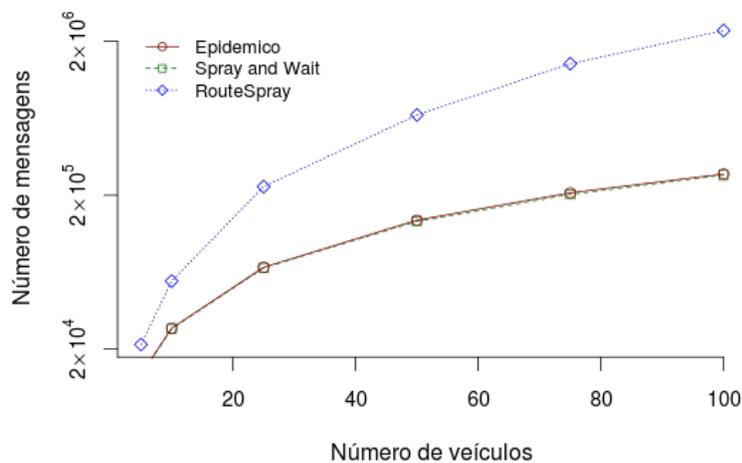


Figura 4. Mensagens enviadas na rede.

5. Conclusões

Roteamento em redes veiculares ainda é um problema em aberto. Diversos algoritmos surgem para cenários e condições específicas. Considerando que em redes VANETs temos recursos que não são presentes em outros tipos de redes, e que com isso conseguimos informações que podem ser utilizadas no roteamento sem custo adicional à rede, foi apresentado neste artigo, o algoritmo *RouteSpray*. Este se mostrou mais eficiente do que os algoritmos até então apresentados pela comunidade científica para roteamento veicular, utilizando como base a rota dos veículos.

Apesar do desafio de rotear pacotes por redes altamente dispersas, o que é comum em redes VANETs, o algoritmo *RouteSpray* garantiu uma boa taxa de entrega de mensagens, superando algoritmos cuja eficiência já foi comprovada pela comunidade científica. Além da boa taxa de entrega de mensagens, o algoritmo *RouteSpray* demandou pouca utilização de espaço de armazenamento, podendo ser utilizado em dispositivos com recursos restritos.

O algoritmo *RouteSpray* tem uma boa aplicação em redes onde se conhece a rota dos veículos. Um exemplo prático de uma boa utilização seria em empresas de transporte, como empresas de ônibus, de táxis ou transportadoras. O algoritmo *RouteSpray* oferece a possibilidade de comunicação dinâmica mesmo em situações não rotineiras onde o roteamento programado falharia, como quando ocorrem atrasos por razão de mudanças no trânsito, pneu do veículo furado, etc.

Referências

- Allal, S. and Boudjit, S. (2012). Geocast routing protocols for vanets: Survey and guidelines. In *2012 Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS)*, pages 323–328.
- Gamess, E., Acosta, L., and Hernandez, D. (2012). Analyzing routing protocol performance versus bitrate in vehicular networks. In *2012 Global Information Infrastructure and Networking Symposium (GIIS)*, pages 1–4.
- Johnson, D. and Maltz, D. (1996). Dynamic source routing in ad hoc wireless networks. In Imielinski, T. and Korth, H., editors, *Mobile Computing*, volume 353 of *The Kluwer International Series in Engineering and Computer Science*, pages 153–181. Springer US.
- Köpke, A., Swigulski, M., Wessel, K., Willkomm, D., Haneveld, P. T. K., Parker, T. E. V., Visser, O. W., Lichte, H. S., and Valentin, S. (2008). Simulating wireless and mobile networks in OMNeT++ the MiXiM vision. In *Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems & Workshops, Simutools '08*, pages 71:1–71:8, ICST, Brussels, Belgium, Belgium. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).
- Lee, K. and Gerla, M. (2010). Opportunistic vehicular routing. In *2010 European Wireless Conference (EW)*, pages 873–880.
- Li, F. and Wang, Y. (2007). Routing in vehicular ad hoc networks: A survey. *IEEE Vehicular Technology Magazine*, 2(2):12–22.
- Perkins, C. and Royer, E. (1999). Ad-hoc on-demand distance vector routing. In *Second IEEE Workshop on Mobile Computing Systems and Applications (WMCSA)*, pages 90–100.
- Silva, M. J. (2012). VeNeM: Vehicular network mobility. Disponível em <https://github.com/badriciobq/VeNeM/>.
- Spyropoulos, T., Psounis, K., and Raghavendra, C. (2008a). Efficient routing in intermittently connected mobile networks: The single-copy case. *IEEE/ACM Transactions on Networking*, 16(1):63–76.

- Spyropoulos, T., Psounis, K., and Raghavendra, C. S. (2008b). Efficient routing in intermittently connected mobile networks: The multiple-copy case. *IEEE/ACM Transactions on Networking*, 16(1):77–90.
- Taysi, Z. and Yavuz, A. (2012). Routing protocols for GeoNet: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 13(2):939–954.
- Tchakountio, F. and Ramanathan, R. (2001). Tracking highly mobile endpoints. In *Proceedings of the 4th ACM International Workshop on Wireless Mobile Multimedia, WOW-MOM '01*, pages 83–94, New York, NY, USA. ACM.
- Toor, Y., Muhlethaler, P., and Laouiti, A. (2008). Vehicle ad hoc networks: Applications and related technical issues. *IEEE Communications Surveys Tutorials*, 10(3):74–88.
- Vahdat, A., Becker, D., et al. (2000). Epidemic routing for partially connected ad hoc networks. Technical report, Technical Report CS-200006, Duke University.
- Varga, A. (1999). Using the OMNeT++ discrete event simulation system in education. *IEEE Transactions on Education*, 42(4):11 pp.–.
- Zhao, J. and Cao, G. (2008). VADD: Vehicle-assisted data delivery in vehicular ad hoc networks. *IEEE Transactions on Vehicular Technology*, 57(3):1910–1922.