

Veer: A Trajectory-Based Peer Selection Algorithm for Networks of Vehicles

Sávio Rodrigues Cavalcanti*, Miguel Elias Mitre Campista*, Fehmi Ben Abdesslem[◇],
Luís Henrique Maciel Kosmowski Costa*, Marcelo Dias de Amorim[◇],
and Otto Carlos Muniz Bandeira Duarte*

* COPPE/POLI – Universidade Federal do Rio de Janeiro, Brazil

[◇] LIP6/CNRS – UPMC Univ Paris 06, France

Emails: {savio,miguel,luish,otto}@gta.ufrj.br ; {fehmi,amorim}@rp.lip6.fr.

Abstract— We propose Veer, a peer selection algorithm for file exchange in vehicular ad hoc networks. The innovative aspect of Veer is that it relies on embedded GPS devices to determine individual mobility patterns and, consequently, to help select the peers that increase the probability that a file transfer succeeds. Transfer opportunities are computed in advance based on an estimation of the contact time window between a node and its selected peers. We show that such an opportunistic one-hop transfer mechanism leads to better results than traditional multi-hop file transfers both in terms of goodput and delivery ratio.

I. INTRODUCTION

One of the challenging goals of vehicular ad hoc networks (VANETs) is the support of legacy applications such as peer-to-peer file sharing. In file sharing applications, the contact time between two peers must be long enough so that they can establish a connection and transfer the required amount of data. The difficulty here is that file sizes might vary within a wide range of values – this implies that we must first evaluate if a contact will be long enough to support a transfer. Note that in the particular context of wireless networking, such a contact time can be very short, as it refers to the time peers are within the communication range of each other. The contact time depends on many parameters, such as the relative speed of the peers, the wireless technology used, or the environmental conditions [1]. On the other hand, if communication protocols are properly tuned, even very short contact times can be used to transfer a few megabytes of data, using off-the-shelf wireless equipment [2].

Routing protocols for VANETs have been proposed in the literature, but in general they rely on existing solutions issued from research on mobile ad hoc

networks (MANET) [3], [4]. These protocols present interesting properties (e.g., management of disconnected scenarios), but face some scalability problems in highly dynamic situations. With proactive protocols, a storm of control messages is required in order to maintain routing tables. Although reactive protocols are preferred in highly dynamic scenarios, breaking paths causes excessive broadcasting for new routes to be discovered.

Because of the potentially short contact windows and the specificities of file sharing applications, relying on existing solutions is not sufficient. For example, as a source does not know the contact duration, a large-file transfer may be abruptly canceled. More specifically, we need a solution that tackles the two following goals *at the same time*:

- **Expected contact time.** The strategy used by a peer to transfer data to a neighbor should rely on the expected time window these two peers remain in contact. On the one hand, too short contacts might be insufficient or should be used only for transferring small packets. On the other hand, longer contacts are of course more appropriate for larger messages. Having the possibility to estimate future contacts would be of great utility to properly schedule transfers and thus increase the overall capacity of the system.
- **Avoid multihop communication.** Finding a long enough contact is by itself a difficult task, especially in highly dynamic networks such as VANETs. The use of multihop-based routing protocols would require a product of contacts that would depend on the path length (in number of hops). We advocate, whenever possible, the avoidance of multihop communications; instead, nodes should rely on their

mobility to decrease distance (ideally to become direct neighbors) before communicating.

In this paper, we propose Veer, a peer selection algorithm that awaits one-hop contacts to avoid multi-hop data transfers and improve file download performance. The idea behind Veer is to benefit from information concerning the network to select the set of peers that increases the number of successful downloads. Veer relies on an estimation of the trajectory of the nodes, contrary to existing approaches that trigger communications only when nodes meet [5], [6].

We compared Veer with the traditional multihop algorithm. Our results show that Veer provides higher packet delivery rate and allows more files to be transferred successfully. For different values of network density, Veer achieves a delivery ratio that is up to 25% higher than the multihop approach. In general, the advantages are more important for larger files and for denser networks. More interestingly, we observe that, although Veer looses in terms of throughput, it leads to higher goodput rates (i.e., only when the whole file is correctly received). This result indirectly shows that Veer achieves better resource usage.

The remainder of this paper is organized as follows. Section II presents the algorithms used for peer selection in Veer. Section III presents simulation results comparing Veer with the traditional multihop approach. Finally, Section V concludes the paper and identifies open research topics related to Veer.

II. VEER: PEER SELECTION BASED ON TRAJECTORIES

The original aspect of Veer is that it relies on the nodes' ability to estimate their trajectories. It is important to underline that we do not focus on trajectory computation. Instead, we consider that nodes can determine where they will be at specific points in time until their final destination, and that this trajectory can be informed to other nodes. This is a realistic assumption since nowadays more and more vehicles are being equipped with GPS devices. Veer exploits these trajectories by previewing contacts between nodes to improve the success rate of file transfers between peers.

We define the operation of Veer by considering the basic operation of a major P2P file transfer system (Gnutella). A node searching for a specific file sends a file query to the P2P network. Nodes that have the file respond with a query hit message. Then, the requesting node selects one or more peers to download the file from. Contrary to traditional P2P systems, the requesting node

TABLE I
MESSAGES IN VEER.

| Message | Purpose | Fields |
|----------|-------------------------------------|---|
| query | File request | File id |
| queryhit | Reply to query if file is available | File id, file size, and node trajectory |
| sendme | File transfer request | File id |

in Veer only selects peers that it will meet in a near future (based on the estimated contact). This does not imply, however, that the requesting node will always be precise when computing the contact time. The key idea is to know in advance for whom to request the file in a given interval of time to improve the delivery rate and, furthermore, the network goodput. This approach avoids multihop transfers, which have shown poor performance even in static scenarios [7]. Additionally, as we will see in Section III, this drastically reduces network utilization.

We define the trajectory of a vehicle n_v as a vector of tuples, denoted by:

$$\mathcal{T}(n_v) = [(t_0, X_0, Y_0, d_0); (t_1, X_1, Y_1, d_1); \dots], \quad (1)$$

where t_i represents the time a vehicle enters a street, (X_i, Y_i) are the geographical coordinates of the point where the vehicle entered the street, and d_i indicates the direction the vehicle travels in this street. More specifically, coordinates (X_i, Y_i) represent the initial point of the street segment the node travels, beginning at time t_i , until the node reaches point X_{i+1}, Y_{i+1} , at time t_{i+1} , when it enters another street in the direction d_{i+1} .

We show in Table I the three types of messages defined in Veer. The steps are also illustrated in Fig. 1. To request a file, the requesting node, n_{req} , first floods the network with a `query` message. Different nodes in the network may have the requested file. Without loss of generality, consider a node n_{resp} receiving the `query` message. Algorithm 1 describes the steps node n_{resp} performs upon reception of the `query` message. First, n_{resp} checks if it has file A . If yes, it becomes a node that is *candidate* (n_{cnd}) for transferring the file A to n_{req} . Then, it obtains the trajectory to its final destination and fills the vector $\mathcal{T}(n_{cnd})$, which will be sent along with a `queryhit` message to n_{req} .

The requesting node n_{req} may receive `queryhits`

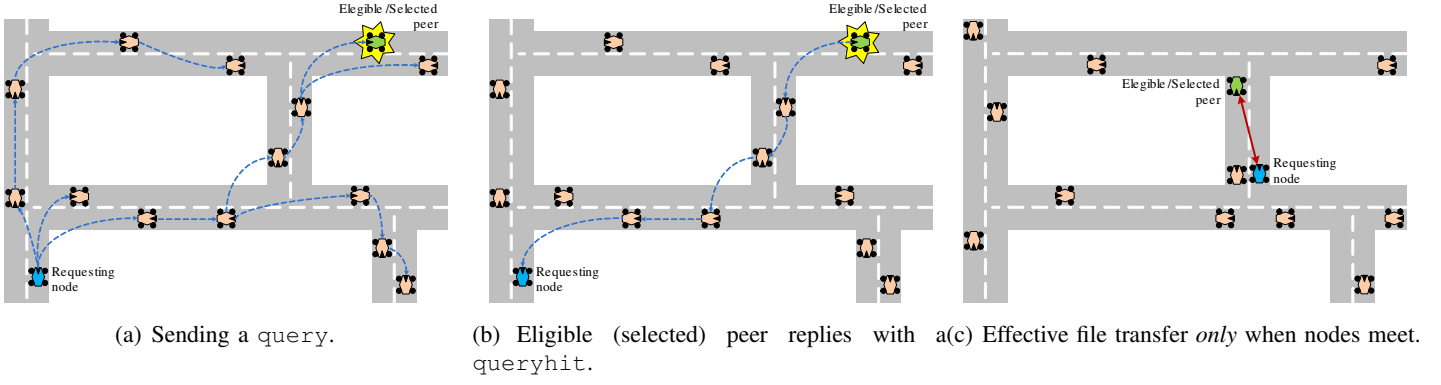


Fig. 1. Example scenario of Veer's operation.

Algorithm 1 n_{resp} received query of file A from n_{req} .

```

if ( $n_{resp}$  owns file  $A$ ) then
  Become  $n_{cnd}$ 
  Obtain  $(\mathcal{T}(n_{cnd}))$  to final destination
  Send queryhit( $\mathcal{T}(n_{cnd})$ , file-size( $A$ )) to  $n_{req}$ 
else
  Forward query to neighbors
end if

```

from one or more nodes. For each queryhit received, node n_{req} checks if a contact with node n_{cnd} may happen, based on $\mathcal{T}(n_{cnd})$ and on its own trajectory $\mathcal{T}(n_{req})$ (cf., Algorithm 3 later). Algorithm 2 describes the steps performed by n_{req} . Comparing the trajectory of the candidate node and its own, n_{req} determines if there are intersections and a contact is expected to happen. If so, it evaluates whether the contact time will last long enough for the file transfer, based on the file size. If all conditions are met, n_{cnd} becomes a selected peer, called n_{sel} , and n_{req} schedules a download of the requested file for the expected contact time. The contact time is also computed by the Search_Contact procedure, but was omitted for sake of simplicity.

Figure 2(a) illustrates all the steps performed by node n_{resp} upon reception of a query message, whereas Figure 2(b) illustrates the steps performed by the requesting node n_{req} upon reception of a queryhit message.

Algorithm 3 is used in the Search_Contact procedure of Algorithm 2 to check if nodes n_{req} and n_{cnd} will meet, given their trajectories, and if the contact time is enough to transfer file A . First, the algorithm checks if nodes n_{req} and n_{cnd} share the same X or Y coordinates at the same time, which means that they may be in the same street. Then, the algorithm checks if the nodes are within radio range (R) of each other, as well as if

Algorithm 2 n_{req} received queryhit from n_{cnd} .

```

Obtain  $(\mathcal{T}(n_{req}))$  to final destination
if Search_Contact( $\mathcal{T}(n_{req})$ ,  $\mathcal{T}(n_{cnd})$ )  $\geq t_{download}(A)$ 
then
   $n_{cnd}$  becomes  $n_{sel}$ 
  if ( $\nexists$  schedule between  $n_{req}$  and  $n_{sel}$  for the file  $A$ )
  then
    Schedule file transfer from  $n_{sel}$ 
  end if
end if

```

they are moving in the same direction. For all the other combinations, the nodes are not in the same street even though the X or Y coordinate is the same. Finally, the algorithm computes the estimated contact time between the vehicles by checking in their trajectories when they will be out of range.

Algorithm 3 Search_Contact($\mathcal{T}(n_{req})$, $\mathcal{T}(n_{cnd})$).

```

if ( $X_{req} = X_{cnd}$ ) OR ( $Y_{req} = Y_{cnd}$ ) then
  if distance( $n_{req}$ ,  $n_{cnd}$ )  $\leq R$  then
    if same direction OR opposite direction then
      return Contact_Time( $n_{req}$ ,  $n_{cnd}$ )
    end if
  end if
end if
return 0

```

Algorithm 3 may return zero, meaning that there will be no contact between the requesting and eligible nodes. If after running Algorithm 2 for all received queryhits, the node could not find an eligible node with whom a contact will occur, then it selects a peer to try a multi-hop transfer. If, on the other hand, Veer

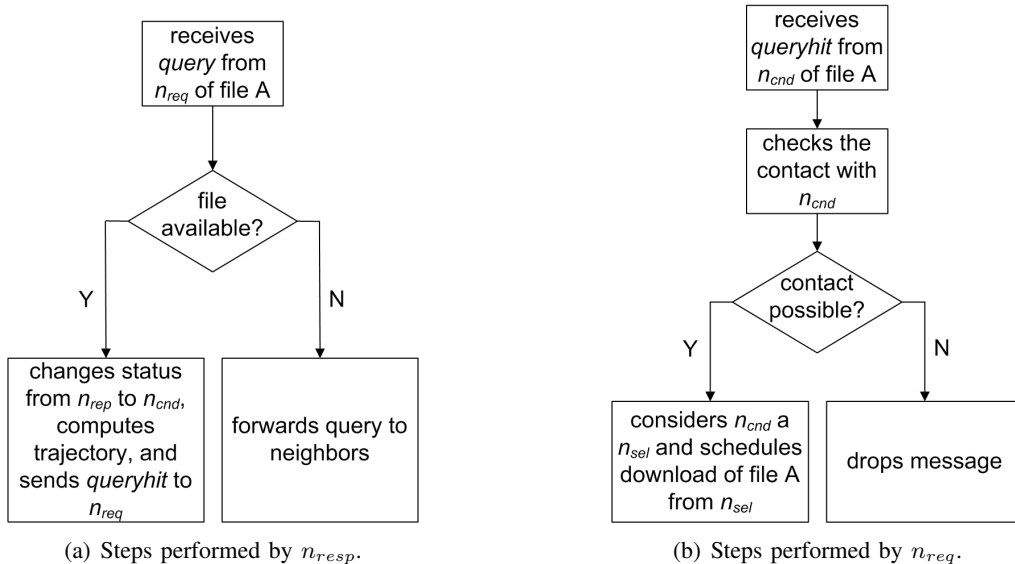


Fig. 2. Operation of n_{resp} and n_{req} nodes upon reception of query and queryhit messages, respectively.

succeeds to schedule a file transfer using a one-hop contact, it has only used multi-hop transmission for the query and queryhit control messages. As query messages are flooded in the network, Veer only relies on a routing protocol for the queryhit messages. As for file transfers, they are performed through a direct link when in range. As a consequence, no specific routing protocol is required whenever Veer is able to find one-hop contacts.

III. ANALYSIS

We simulated Veer on the NS-2.31 [8] network simulator. The simulation scenario consists of a square of size 650 m divided into 25 blocks, where the number of vehicles ranges from 30 to 120. There is no auxiliary communication infrastructure. We assume that all streets are two-way and traffic lights are placed at the corners. Node speed is variable and limited to 40 km/h. Mobility scenarios are generated using GHOST [9], which provides a mobility model where the nodes are confined to straight lines representing the streets of a city. In real life, the trajectories of the node will be obtained from a GPS device, as mentioned in Section II. In the simulations, we compute the vector of tuples which describes the trajectory of a node from the mobility scenario produced with GHOST. For the purposes of this paper, we consider that trajectories do not change during the simulations.

Each node starts with 6 files. There are in total 26 different files (noted from “A” to “Z”), which are uniformly distributed among nodes (note that different nodes may have the same file). File sizes range from

500 kB to 3 MB. Each node requests 3 files at random. Each simulation run lasts for 300 s. Each file request is randomly triggered at a time between 0 and 290 s, following a uniform distribution. Note that, instead of files, in a real implementation the transfer unit may be file segments (depending on the application protocol used by the peers).

We compare Veer with the traditional multihop based algorithm (i.e., file transfers begin as soon as the first reply is received, using multi-hop transmission). For each simulation setup, we make 30 simulation runs. The same mobility scenario is used for Veer and the reference multi-hop algorithm. We simulate an IEEE 802.11b wireless network and nodes transfer files using FTP connections.

A. Results

We compare the performance of Veer with the multi-hop reference algorithm. We evaluate three performance metrics: packet delivery ratio, network aggregated throughput, and network aggregated goodput. The packet delivery ratio is measured as the ratio between packets sent and packets received at the application level. We define the aggregated throughput as the amount of bits received by all nodes at the application level over the simulation time, measured in Mbps. The aggregated goodput is also measured in Mbps, and indicates the amount of bits received over the simulation time, only when the whole file is correctly received. For both algorithms, a node sends only one request per file.

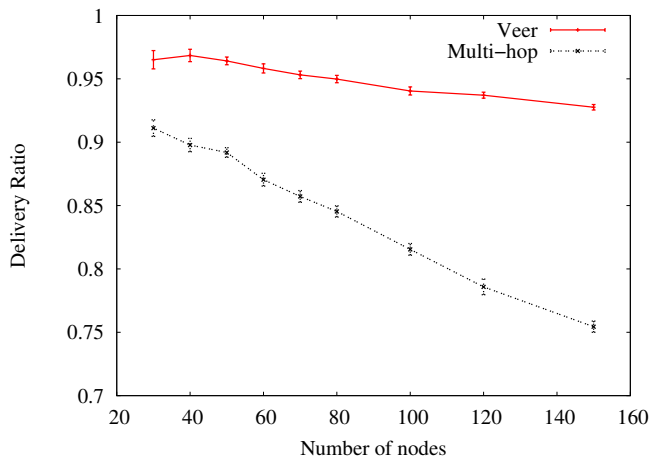


Fig. 3. Delivery ratio.

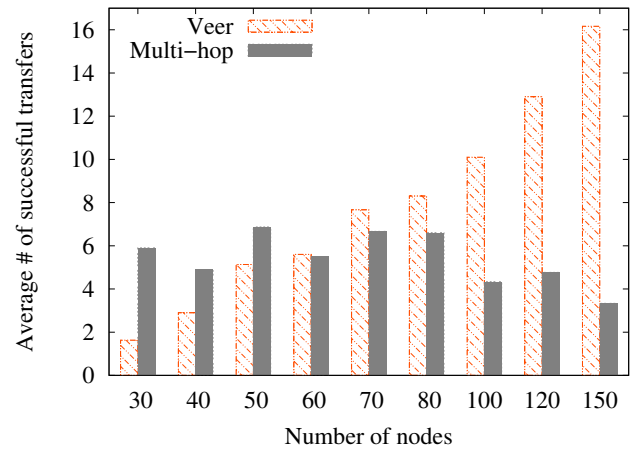


Fig. 5. Average number of successful transfers with files from 1 MB to 2 MB.

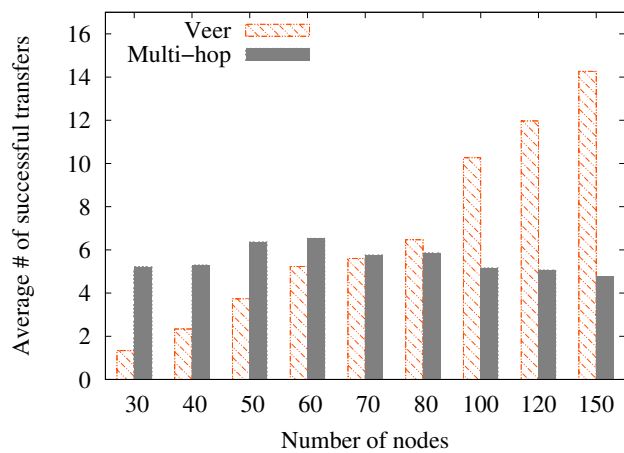


Fig. 4. Average number of successful transfers with files smaller than 1 MB.

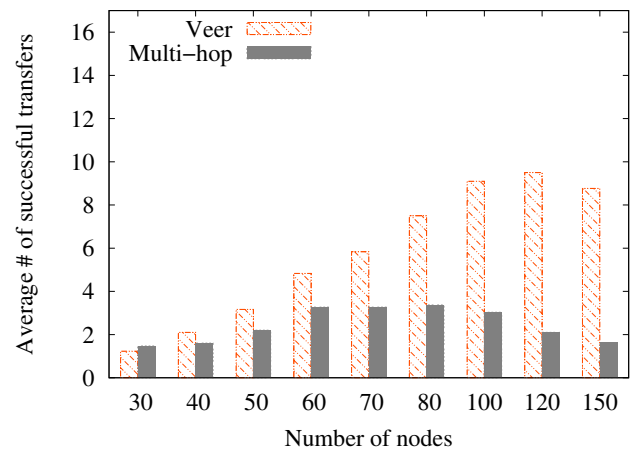


Fig. 6. Average number of successful transfers with files larger than 2 MB.

Fig. 3 compares the packet delivery ratio obtained by Veer with the one obtained with the multi-hop algorithm. Veer's delivery ratio is above 92% for all network sizes. As the number of cars increases and the VANET becomes denser, we observe a small decrease in the delivery ratio. With the reference algorithm, on the other hand, the delivery ratio is always lower than the one obtained by Veer and decreases rapidly as the number of vehicles increases. With more nodes in the same area, there are more interference and medium contention. When the file transfer is performed through a multi-hop path, interference and contention become more severe because the number of medium accesses per packet grows with the path length.

The packet delivery ratio is not the most important metric given that the P2P application of interest is file transfer. Therefore, we have also analyzed the average

number of files successfully transferred as well as the corresponding goodput.

Figs. 4 to 6 present the average number of files successfully transferred in all simulation runs. Files are grouped based on their size; we investigate the impact of the file size on a specific scenario/algorithm. Results show that, by using an opportunistic approach, Veer performs better than multi-hop communication when the number of vehicles is larger than 80, for any file size. The reason is that, with more nodes, the probability for a vehicle to encounter another one to schedule a one-hop file transfer is higher. This is particularly interesting in urban scenarios, which are typically dense. This result indicates that by scheduling one-hop transfers, the contact time between communicating peers tends to be enough to transfer larger files (or file segments) as

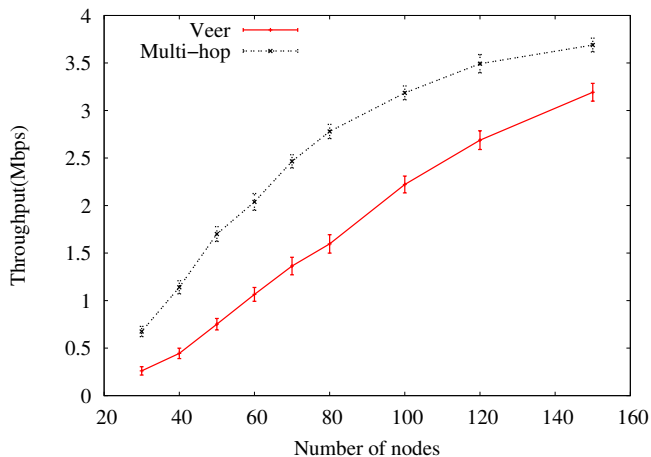


Fig. 7. Throughput.

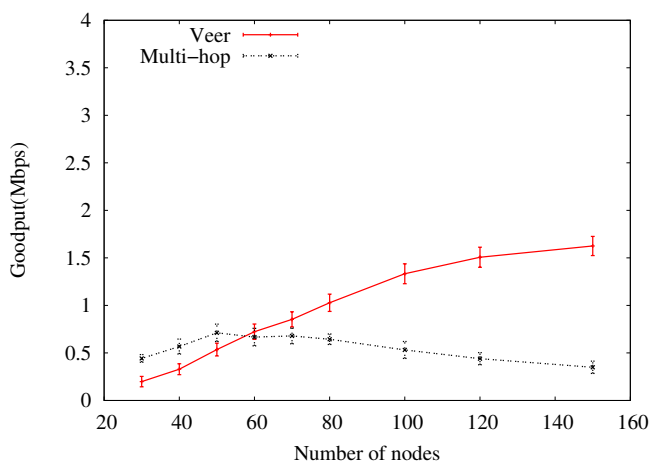


Fig. 8. Goodput.

compared to the lifetime of a multi-hop path. However, multi-hop file transmissions start as soon as an end-to-end path is found. This strategy is efficient for smaller files (or file segments) because it can establish routes more often, regardless of their duration. Clearly, the tradeoff incurred by Veer's one-hop file transfers is increased latency, as nodes may have to wait for one-hop contacts. Nevertheless, latency is not the main issue for P2P file transfers. The download of a large file may take hours using current Internet P2P systems. On the other hand, Veer provides much larger bandwidth efficiency, which is currently an important issue for mobile wireless networks.

Figs. 7 and 8 present the aggregated throughput and the aggregated goodput obtained with Veer and the reference algorithm, respectively. Fig. 7 shows that the network does not saturate with the number of nodes. We

TABLE II
EFFICIENCY (GOODPUT/THROUGHPUT).

| # of Nodes | Veer | Multi-hop |
|------------|--------|-----------|
| 30 | 76.18% | 50.55% |
| 40 | 73.89% | 54.85% |
| 50 | 71.21% | 54.23% |
| 60 | 67.91% | 51.60% |
| 70 | 62.58% | 48.27% |
| 80 | 64.37% | 46.69% |
| 100 | 60.01% | 44.30% |
| 120 | 56.06% | 41.02% |
| 150 | 50.92% | 38.88% |

observe that sending through multiple hops leads to a higher throughput than Veer, because when a requested file is found the transfer begins sooner. On the other hand, as far as the goodput is concerned (Fig. 8), Veer shows a similar efficiency from 60 to 70 nodes and outperforms the reference algorithm for more than 80 nodes. Note that, starting at 50 nodes, the increasing interference in the network considerably degrades multi-hop communication. Another consideration is that the percentage of goodput in relation to throughput for multi-hop transmissions is lower than Veer, as we can see in Table II. Therefore, in this kind of VANET scenario, using the one-hop file transfers computed in advance is better than using multi-hop paths in terms of network resources usage.

IV. RELATED WORK

Peer selection mechanisms were also proposed with safety applications in mind. Within the Cartalk [6] project, the development of delay-intolerant systems was investigated to support traffic alert systems. The project used as example a system where in case of traffic jam, vehicles behind the originating vehicle receive warning messages from it, whereas for vehicles preceding the originating vehicle, or travelling in the opposite direction, the warning is meaningless given their trajectory and, therefore, no message is sent to those vehicles.

Routing in vehicular ad hoc networks is an open research issue related to our work. Naumov *et al.* [10] investigated how two legacy ad hoc routing protocols, AODV (Ad Hoc on demand Distance Vector [11]) and GPSR (Greedy Perimeter Stateless Routing [12]), perform for VANETs. Their results show that when AODV is used for dense scenarios, half of AODV's route request messages were lost due to collisions. On the other hand, GPSR has suffered from inconsistencies in the neighbor

information tables, leading to poor performance in terms of throughput. Their conclusion is that both protocols presented performance issues used in urban scenarios, even if AODV reached twice the packet delivery rate as GPSR.

Ducourthial *et al.* proposed a routing mechanism for VANETs which is based on conditional transmissions, called HOP [13]. Using simulation, they have compared HOP's performance in terms of data delivery rate against other 5 routing protocols: OLSR (Optimized Link State Routing Protocol), Fast OLSR, AODV, LBM (Location-Based Multicast), and GAMER (Geocast Adaptive Mesh Environment for Routing). OLSR and Fast OLSR, which are link-state protocols, have shown the worst performance because of the rapid topological changes, which cause inconsistencies in the routing tables. AODV performed better than OLSR and Fast OLSR but was affected by larger distances between vehicles, whereas LBM and GAMER have faced problems with node mobility, since both optimize message delivery based on specific areas. On the other hand, as HOP does not need neighbor information or control messages, it performed better than the other protocols.

Chen and Cai [5] proposed the Local Peer Group (LPG) mechanism which organizes neighbor vehicles into groups. The objective is to reduce the message delivery latency inside a group, optimizing the network for safety warning applications. The organization of vehicles into groups also allows that the coverage and direction of warning messages be controlled, for example, selecting the vehicles which will receive an emergency alert according to their position. Two types of group organization were proposed. The first one named static uses the vehicle's location obtained from a GPS device to partition the streets according to the postal code, for example. On the other hand, the dynamic group organization divides groups according to the vehicles which are within radio communication range of each other. This strategy also avoids reply implosion because the number of responding nodes is limited [14].

Within the field of Delay and disruption Tolerant Networks (DTN), some work explore the contact history between nodes to make routing decisions based on the probabilistic prediction of future contacts [15], [16]. The use of such technique, in case exact trajectories are not available, deserves investigation.

V. CONCLUSIONS

In this paper, we proposed Veer, a peer selection algorithm to perform file transfers in VANETs. The

originality of our approach is that it proposes a different strategy to deal with mobility, namely, to use a prediction of vehicles' trajectories to optimize the performance peer-to-peer applications. Veer schedules one-hop transfers with peers that have similar trajectories during a given interval of time. We showed that such an opportunistic approach leads to better results than a simple mechanism based on multi-hop transfers, in terms of packet delivery rate, goodput, and scalability.

Future work includes the design of a robust mechanism to control packet losses. We will also study whether an adaptive scheme combining Veer with multi-hop transmissions could be better than using only Veer when there is temporary low mobility, e.g., in case of traffic jam. We also plan to consider abrupt changes in the behaviors of the vehicles (due for instance to change of route, a red traffic light, or an accident).

VI. ACKNOWLEDGMENTS

The authors would like to thank FINEP, FUNTTEL, CNPq, CAPES, and FAPERJ for partially funding this research.

REFERENCES

- [1] D. Hadaller, S. Keshav, T. Brecht, and S. Agarwal, "Vehicular opportunistic communication under the microscope," in *Proceedings of the 5th International Conference on Mobile Systems, Applications and Services (ACM MobiSys)*, San Juan, Puerto Rico, June 2007, pp. 206–219.
- [2] M. G. Rubinstein, F. B. Abdesslem, S. R. Cavalcanti, M. E. M. Campista, R. S. Alves, L. H. M. K. Costa, M. D. Amorim, and O. C. M. B. Duarte, "Measuring the capacity of in-car to in-car vehicular networks," *IEEE Communications Magazine*, vol. 47, no. 11, pp. 128–136, Nov. 2009.
- [3] N. Wisitpongphan, F. Bai, P. Mudalige, V. Sadekar, and O. Tonguz, "Routing in sparse vehicular ad hoc wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 8, pp. 1538–1556, oct 2007.
- [4] T. Taleb, E. Sakhaee, A. Jamalipour, K. Hashimoto, N. Kato, and Y. Nemoto, "A stable routing protocol to support ITS services in VANET networks," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 6, pp. 3337–3347, Nov. 2007.
- [5] W. Chen and S. Cai, "Ad hoc peer-to-peer network architecture for vehicle safety communications," *IEEE Communications Magazine*, vol. 43, no. 4, pp. 100–107, Apr. 2005.
- [6] Daimler Chrysler, "Calling all cars," Project CarTALK 2000, Tech. Rep., 2001, <http://www.cartalk2000.net/>.
- [7] J. Camp and E. Knightly, "The IEEE 802.11s extended service set mesh networking standard," *IEEE Communications Magazine*, vol. 46, no. 8, pp. 120–126, Aug. 2008.
- [8] T. V. Project, "The network simulator (NS-2)," Available: <http://www.isi.edu/nsnam/ns/> [Online], 2010.
- [9] F. Legendre, V. Borrel, M. D. de Amorim, and S. Fdida, "Modeling mobility with behavioral rules: the case of incident and emergency situations," in *Asian Internet Engineering Conference (AINTEC)*, Bangkok, Thailand, Nov. 2006.

- [10] V. Naumov, R. Baumann, and T. Gross, "An evaluation of inter vehicle ad hoc networks based on realistic vehicular traces," in *ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, May 2006, pp. 108–119.
- [11] C. E. Perkins, E. M. Belding-Royer, and R. S. Das, *Ad Hoc On-Demand Distance Vector Routing*, Request for Comments: 3561, July 2003.
- [12] B. Karp and H. T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in *ACM International Conference on Mobile Computing and Networking (MobiCom)*, Aug. 2000, pp. 243–254.
- [13] B. Ducourthial, Y. Khaled, and M. Shawky, "Conditional transmissions: Performance study of a new communication in VANET," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 6, pp. 3348–3357, Nov. 2007.
- [14] A. T. A. Gomes, A. Ziviani, L. S. Lima, M. Endler, and G. Chelius, "Mitigating reply implosions in query-based service discovery protocols for mobile wireless ad hoc networks," in *International Conference on AD-HOC Networks and Wireless (ADHOC NOW)*, Sept. 2008, pp. 29–42.
- [15] E. C. R. de Oliveira and C. V. N. de Albuquerque, "NECTAR: a DTN routing protocol based on neighborhood contact history," in *ACM Symposium on Applied Computing*, Mar. 2009, pp. 40–46.
- [16] A. Lindgren, A. Doria, and O. Scheln, "Probabilistic routing in intermittently connected networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 7, no. 3, pp. 19–20, July 2003.