Disseminating Real-Time Traffic Information in Vehicular Ad-Hoc Networks

Ting Zhong, Bo Xu and Ouri Wolfson

Abstract—**In this paper we propose an algorithm for disseminating reports about real-time traffic conditions in vehicular ad-hoc networks. Using this method, each vehicle makes local decision on when to disseminate reports, how many to disseminate, and which reports to disseminate. In order to deal with the bandwidth and memory constraints, reports are prioritized in terms of their value, as reflected by supply and demand. We compare the proposed algorithm with Grassroots, an existing VANET dissemination algorithm. The comparison is based on simulation of vehicle mobility in a real road network and of the 802.11 protocol. The simulation results show that the proposed algorithm outperforms Grassroots in the environments where VANET dissemination is challenging.**

I. INTRODUCTION

Traffic congestion has become part of daily life in many places. Vehicle navigation systems assist driving by combining the use of digital maps with the location information reported by the GPS receiver. However, in basic vehicle navigation systems, information about real-time traffic conditions is not available, and therefore the route planning does not adjust to avoid congestions.

Existing commercial systems for collecting and disseminating traffic information (e.g., Traffic.com [1], GCM Travel [3]) are implemented by a centralized architecture. These systems usually use roadside sensors to get the traffic information and send the information to central servers. Thus, users are able to select alternative routes in order to avoid congested streets. However, these existing systems tend to cover selected highways where speed sensors are deployed, while leaving out a major fraction of roadways. The main factor that prevents these systems from covering the entire road network is the cost involved to deploy sensors to cover the entire road network.

With a GPS receiver and a digital map installed, a vehicle is able to keep track of its location and to determine its travel time for each road segment recently visited. By sharing the travel time information in a peer-to-peer fashion, vehicles can be aware of the traffic situation of the road network and accordingly change their routes to avoid the congestion. A vehicular ad-hoc network (VANET) is a set of vehicles that communicate via short-range wireless technologies such as IEEE 802.11 and DSRC. Each vehicle m participating in the VANET periodically produces reports regarding the traffic condition it is experiencing. With the VANET, the reports are disseminated to neighboring vehicles or to remote vehicles by multi-hop transmission relayed by the intermediate vehicles. Compared to the centralized, the VANET approach has the following advantages:

1. Due to the fact that short-range wireless networks utilize the unlicensed wireless spectrum, communication in a VANET is free. In addition, there is no cost involved in setting up and maintaining the fixed infrastructure server.
2. The VANET approach enables vehicles to collect traffic conditions of both highways and other roadways.

There are two paradigms to conduct reports dissemination in a VANET, namely structure-full and structureless. In structure-full dissemination, a routing structure is imposed and maintained among the vehicles (e.g., [11]). Structure-full dissemination may be very ineffective in a highly mobile environment, as the routing structure quickly becomes obsolete (see [10]).

In structureless dissemination, no routing structures are maintained; the intermediate vehicles save reports to their local databases and later transfer these reports. In the literature this paradigm is also called structureless gossiping, epidemic, or store-and-forward dissemination. The problem with store-and-forward is that the reports that need to be stored and forwarded by a vehicle may exceed its storage and bandwidth capacities. The algorithm we propose in this paper addresses this problem by ranking of reports, so that the most relevant reports are transmitted and saved to the local database.

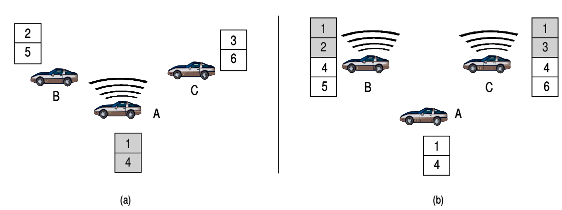

Fig. 1.1. Rank-based store-and-forward. (a) Vehicle *A* broadcasts its top two reports which are reports 1 and 4. (b) After receiving from *A*, vehicle *B* incorporates the received reports, re-ranks, and broadcasts the top two (shadowed). The same for *C*.

At a high level, our rank-based store-and-forward method works as follows (see Figure 1.1). Each vehicle *O* periodically selects the *k* most relevant reports in its local database, and

This work was supported by US NSF Grants DGE-0549489, 0513736, 0326284, and China NSFC 60573129, 60473090.

Ting Zhong is with the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, Sichuan, 610054 China. She is presently doing research in the Department of Computer Science, University of Illinois at Chicago, IL 60607, USA (phone: 312-996-4881; e-mail: tzhong2000@gmail.com).

Bo Xu and Ouri Wolfson are with the Department of Computer Science, University of Illinois at Chicago, IL 60607, USA (e-mail: {boxu,wolfson}@cs.uic.edu).

broadcasts (i.e. forwards) them to its neighbors (i.e. the vehicles that are within the transmission range of *O*). Upon receiving the broadcast from *O*, each neighboring vehicle incorporates the received reports into its own local database, and subsequently broadcasts the top *k* reports. Thus reports transitively spread across vehicles. Similarly, when the local database size is insufficient to store all the received reports, only the most relevant reports are saved.

The fundamental components for the rank-based store-and-forward are the following:

1. Demand, as introduced in [12], indicates the relevance of the report to vehicles' decision making such as travel-route planning.

2. Supply, as introduced in [13], indicates how many vehicles already have the report. The higher the supply of a report, the lower its relevance. [13] proposes a machine learning algorithm for the estimation of supply.

In this paper we integrate the above components into the *TrafficInfo* algorithm that ranks reports based on supply and demand, and each transmission communicates the "right" amount of information.

In order to evaluate the proposed TrafficInfo algorithm, we experimentally compared it with Grassroots [5], an existing structureless algorithm for reports dissemination in VANET's. In Grassroots, a report is flooded by its producer. The comparison uses the STRAW/SWANS simulation test-bed ([14, 7]). The STRAW/SWANS test-bed simulates the detailed procedure and factors of 802.11 communication and the vehicle traffic mobility.

We propose a novel metric, called the difference in knowledge (DIK), as the performance measure. The DIK measures the difference between the actual traffic condition on a road segment and that known to a vehicle via the VANET dissemination. In the paper we argue that DIK combines the two metrics commonly used for data dissemination, namely throughput and response time.

In summary, the main contributions of this paper are as follows. 1) We propose a rank-based store-and-forward algorithm (TrafficInfo) for disseminating the real-time traffic information in a VANET. 2) We compare TrafficInfo with an existing VANET dissemination algorithm. 3) The comparison uses a novel performance metric which combines the throughput and response time metrics.

The rest of the paper is organized as follows. Section II introduces the model. Section III describes the TrafficInfo algorithm. Section IV compares TrafficInfo with Grassroots. Section V discusses the performance metric and variants of TrafficInfo. Section VI discusses relevant work, and section VII concludes the paper.

II. THE MODEL

A. Digital map

The system consists of a set of vehicles moving on a road network. For the purpose of route planning, each vehicle *O* has a *digital map* of the road network, denoted by DM_O. The digital map is organized by road segments, where a road segment is a stretch of a road between two successive exit points (junction, exits, etc). Each road segment has a unique identifier. For each road segment *s*, the digital map stores three attributes:

1. The identifier of *s*;

2. The coordinates of the endpoints of *s*;

3. The estimated travel time of *s*. Initially, i.e., at the time when *O* enters into the system, the estimated travel time of *s* is set to be the free-flow travel time of *s*. The free-flow travel time is the travel time when *s* is traveled through with its speed limit.

B. Travel-time report and reports database

Each vehicle *O* is equipped with a GPS receiver. Every time *O* travels through a road segment *s* and reaches the end of it, *O* produces a *travel-time report*, or *report*, regarding the travel time experienced by *O* on *s*. The report contains the following attributes:

1. The identifier of s;

2. The experienced travel time, i.e., the travel time experienced by *O* on *s*;

3. The timestamp of s, i.e., the time when the report is produced.

O disseminates the produced report to other vehicles to share its travel time experience. Each vehicle stores its produced and received reports in the *reports database*. The reports database of vehicle *O* is denoted DB_O. DB_O can hold at most F_O reports. Denote by $R.S$ the road segment reported by a report *R* and by $R.T$ the travel time reported by *R*.

C. Updating digital map upon receiving a travel-time report

Upon receiving a report R from another vehicle, *O* updates the estimated travel time of the corresponding road segment in its digital map DM_O. *O* updates the estimated travel time maintained at DM_O according to the following equation ([5]).

$$T(O,R.S) = \alpha \cdot T(O,R.S) + (1-\alpha)\cdot R.T \quad (0 \le \alpha \le 1) \qquad (2.1)$$

$T(O,R.S)$ is the travel time of road segment $R.S$ maintained at *O*'s digital map. In other words, the new estimated travel time is the weighted average of the current estimated travel time and the reported travel time. α is referred to as the *weight coefficient*. In this paper we use equal weights as discussed in [2], i.e., α=0.5.

III. THE TRAFFICINFO ALGORITHM

This section is organized as follows. §III.A presents the principles that are integrated into TrafficInfo dissemination. §III.B discusses reports ranking. §III.C describes the TrafficInfo algorithm.

A. Principles of TrafficInfo Dissemination

The TrafficInfo algorithm is an integration of two mechanisms that enable each vehicle to keep its digital map as up-to-date as possible, under the bandwidth and storage constraints (see Figure 2.1). These mechanisms include:

1. How much to transmit in a broadcast. In TrafficInfo, every time a vehicle *O* travels through a road segment *s* and reaches the end of it, *O* produces a travel-time report for *s* and triggers a broadcast. The broadcast includes the produced report and the reports in *O*'s reports database. Observe that a vehicle may have a lot of reports to transmit in a broadcast but it may not be able to transmit all of them due to bandwidth constraints. How many reports a vehicle can transmit in a broadcast is determined to optimize the utilization of bandwidth. Intuitively, if the transmission size is too small, then the bandwidth is underutilized and the report dissemination suffers. On the other hand, if the transmission size is too big, then many collisions would reduce the number of successfully received reports. Thus there is an *optimal transmission size* that achieves the best tradeoff between the bandwidth utilization and transmission reliability.

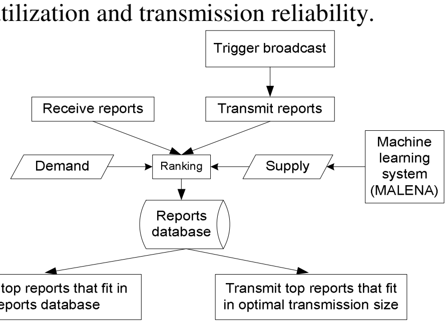

Fig 2.1. Principles of the TrafficInfo algorithm (execution at an individual vehicle)

In this paper the optimal transmission size of each vehicle for each interaction is determined based on the formula developed by [10]. We refer to this formula as the Good Citizen (GC) formula. The GC-formula is applicable to carrier-sense multiple access (CSMA) protocols, such as 802.11 and DSRC. Using this formula a vehicle dynamically adjusts the transmission size *k* based on the length of the time between subsequent broadcasts, such that the overall effective bandwidth is maximized. More precisely, our scheme is as follows. First, we developed the GC-formula that, for a given set of environmental parameters (e.g., available bandwidth, transmission range, vehicle density, transmission frequency), gives the effective throughput, i.e. the number of reports that are correctly received (namely without incurring collisions). By finding the maximum of the GC-formula, we obtain an answer to the following question: assuming that all vehicles broadcast with the same given frequency, what should be the number of reports broadcasted, such that the effective throughput is maximized.

Now clearly, not all vehicles broadcast with the same frequency (e.g., if no new reports are received by a vehicle, and its set of neighbors has not changed, there is no point in it continuously broadcasting the top *k* reports to the same neighbors). Thus each vehicle uses the GC-formula as follows. When it is ready to broadcast, it computes the length *x* of the time period since its last broadcast, and determines based on the GC-formula what is the optimal transmission size *k* for the frequency *1/x*; and it broadcasts *k* reports. In other words, the GC-formula enables each vehicle to maximize the transmission size, while avoiding thrashing (i.e. excessive collisions). Observe that thrashing is caused by the excessive broadcast by all vehicles in the system, a factor out of control of a single vehicle. But the formula enables a vehicle to be a "good citizen", i.e. broadcast only the amount that "is allowed" by the period of time from its last broadcast. A reader is referred to [10] for details.

2. What to transmit during an interaction. Observe that since the amount of transmission is limited, not all the reports in the reports database can be transmitted. Ranking is done to determine which reports to transmit. Intuitively, the rank of a report depends on its demand (how relevant it is to travel-route planning), and supply (how many vehicles already have it). The demand is computed using a spatio-temporal function that decays as the age of the report and the distance of the reported road segment to the vehicle. For the estimation of supply, we use the MALENA algorithm introduced in [13]. Further details of reports ranking are provided in §III.B.

B. *Reports Ranking by Demand and Supply*

In §III.B.1 we define and justify the ranking method. In §III.B.2 we discuss how to estimate the overall supply locally at a vehicle.

1) *The Ranking Method*

The rank of a report *R* at location *p* at time *t* is determined by the following two factors.

1. The *demand of R at location p at time t*, denoted *demand*(*R*,*p*,*t*), represents the relevance of *R* to a vehicle's route planning if the vehicle were at location *p* at time *t*. This relevance depends on two spatial-temporal factors, i.e., the distance from *p* to the road segment reported by *R* (i.e., *R.S*); and on the time that elapsed since *R* was produced. Intuitively, the demand of *R* decays with distance and time. Formally, the demand of *R* is computed as follows.

$$demand(R, p, t) = \frac{1}{c + g} \qquad (3.1)$$

c is the age of *R* (i.e., the time that elapsed since *R* was produced until *t*). *g* is the free-flow travel time along the shortest-distance path from *p* to the middle point of *s*. The purpose of using the travel time along the shortest-distance path rather than the shortest-distance is to make the spatial factor and the temporal factor addable.

2. The *supply of R at time t*, denoted *supply(R,t)*, is the fraction of vehicles in the system at time *t* that have received *R* before time *t*. This number is also a global parameter that is normally unknown by each individual vehicle, but it can be evaluated by the vehicle based on metadata about *R* such as the number of times *R* has been received from other vehicles. The computation of the supply is described in §III.B.2. Formally, the *rank* of *R at location p at time t* is

$$rank(R, p, t) = demand(R, p, t) \cdot (1 - supply(R, t)) \qquad (3.2)$$

Now we justify Equation 3.2. Based on its definition, $demand(R,p,t)$ indicates the relevance of R to a vehicle m at location p at time t, under the condition that R has not been received by m by time t. Based on the definition of $supply(R, t)$, $(1-supply(R,t))$ indicates the probability that R has not been received by m by time t. Thus $demand(R,p,t)\cdot(1-supply(R,t))$ indicates expectation of the relevance of R.

2) Computing Supply by Machine Learning

In this subsection we outline an algorithm, called MALENA (see [13]), for the computation of supply(R) by each vehicle. To introduce the MALENA algorithm, observe that the supply of R depends on attributes of R (e.g. its age), as well as global system parameters such as the turnover rate (i.e. the rate at which vehicles enter and exit the system). The attributes of R that can affect its supply are called *supply indicators*. It can be shown that, unfortunately, no single indicator is a good predictor of supply in all environments. For example, in some environments the intuition that the age of the report is a good predictor of supply is correct whereas in some other environments (e.g. when the turnover is very high) it is not.

MALENA combines various supply indicators in order to estimate the supply. The combination uses machine learning to infer from previously received reports what the indicators of a new report "look like". In other words, it learns the supply based on the supply indicators of reports that it receives. For this purpose, the set of supply indicators of a report R, called the *supply indicator vector* (SIV), are transmitted by each vehicle together with R. Upon receiving R, a vehicle O determines whether or not R is new, and the respective SIV becomes a training example. In other words, O treats itself as an arbitrary vehicle. In this fashion, O progressively collects a training set which improves its learning system. When O ranks reports, the learning system is used to calculate the supply. Furthermore, using a sliding window of examples MALENA can adapt to new environments.

C. Description of the TrafficInfo Algorithm

The execution of TrafficInfo is triggered at a vehicle O by either of the following two events: 1) O reaches the end of a road segment and produces a travel-time report; and 2) O receives a transmission of travel-time reports from another vehicle. The formal description of the TrafficInfo algorithm is given below.

TABLE 3.1 TRAFFICINFO ALGORITHM EXECUTED AT VEHICLE O

Input:	W: The set of travel-time reports produced or received by O; DM_O: The digital map of O; DB_O: The reports database of O; F_O: Size limit of DB_O.
1. For each report in W, update DM_O using Equation 2.1 (see §II.C).	
2. Rank W together with DB_O, using Equation 3.2 (see §III.B); save the top F_O reports in DB_O.	
3. Compute k, the number of reports in the current broadcast, using the GC formula.	
4. Broadcast the top k reports in DB_O.	

IV. EVALUATION OF TRAFFICINFO

In this section we compare TrafficInfo with Grassroots. §IV.A describes the Grassroots algorithm and the case in which reports are not disseminated at all. We refer to this latter case as NonInfo. By comparing with NonInfo, we evaluate the benefit of the VANET dissemination. §IV.B introduces the simulation method. §IV.C presents the simulation results.

A. The Grassroots Algorithm

The Grassroots algorithm (see [5]) is executed by m periodically, at the so-called *maximum dissemination rate*. Upon execution, m examines the travel-time reports it has produced since the last execution of the trigger procedure. Among these, m selects the one for which the difference between the estimated travel time and the travel time actually experienced by m is the maximum. Then a flooding is initiated to disseminate the selected travel-time report (denoted by R). The flooding starts with m broadcasting R to all the neighbors. Each neighbor that receives R in turn immediately rebroadcasts R exactly once. This process is repeated by each vehicle that receives R. We conducted simulations to fine-tune the maximum dissemination rate parameter. It turned out that the performance (defined in IV.B.3) of Grassroots is optimized if m immediately initiates the flooding of a travel-time report after the report is produced by m. In this paper we do so for Grassroots.

B. Simulation Method

1) Simulation Environment

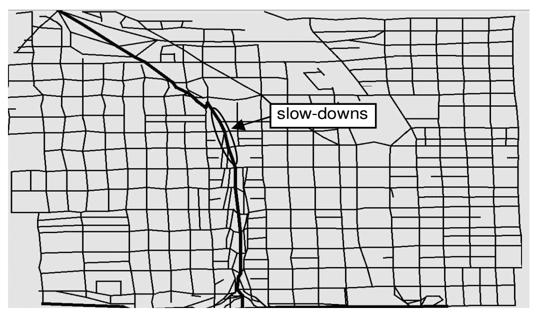

Fig 4.1. Simulated road network: portion of downtown Chicago. The thick curve is highway I-90.

We used STRAW [14] for the simulation of mobility. STRAW provides a microscopic vehicular traffic model on a road network. In the STRAW system, vehicle movement is constrained to roadways defined by real maps. The vehicle mobility is limited according to the speed limit of each road segment, car-following rules, and traffic control mechanisms (e.g., stop signs and timed stoplights).

For all of our experiments, the simulated field is a 3.2km×2.2km region of downtown Chicago (see Figure 4.1). We deployed n vehicles in the region. All the vehicles participate in VANET dissemination. By varying n we varied the density of the VANET network.

Initially, each vehicle m is placed at a random location on the road network, and another random location on the road network is selected to be the destination. m then moves from the origin to the destination. When the destination is reached, another destination is randomly selected. This procedure is repeated until the end of the simulation.

In order to evaluate the effectiveness of each dissemination algorithm, we introduced slow-downs on four high-way segments in the highway I-90 (the circled road segments in Figure 4.1). Specifically, when STRAW simulates the vehicle mobility on each of these high-way segments, the speed limit is set to be 8 kilometers/hour (instead of the default 128 kilometers/hour). All the slow-downs are created at the beginning of the simulation and they last throughout the simulation. The length of each simulation run is 1000 simulated seconds.

The STRAW system uses SWANS (Scalable Wireless Ad hoc Network Simulator) [7] for the simulation of inter-vehicle communication. SWANS implements the IEEE 802.11b Medium Access Control (MAC) protocol and considers detailed communication factors such as the decay of radio signals with increasing distance, signal collisions, and the delay for channel capturing. Using these factors it determines whether each reception succeeds, and how long it takes.

The simulation parameters and their values are summarized in Table 4.1.

TABLE 4.1 SIMULATION PARAMETERS AND THEIR VALUES

Parameter	Unit	Value	Parameter	Unit	Value
Total number of vehicles		100, 300, 500	802.11 bandwidth	bits/second	2M
Transmission range	meter	250	Size of each report	byte	100
Aggregation weight α		0.5	Size limit of reports database	report	200
Slow-down speed	km/hour	8	Length of each simulation run	second	1000

We also tested the cases in which each vehicle allocates only a fraction of the available short-range bandwidth to VANET dissemination (the rest may be used for internet access and videos downloads through the VANET). This fraction also models the decrease of the 802.11 efficiency due to the interference of other wireless technologies that use the same band with 802.11. The results are omitted for space considerations.

2) *Performance Measure*

At the intuitive level, the performance measure is the difference between the travel time maintained by each vehicle for each road segment and the actual travel time of that road segment. In the following we first formally define the actual travel time of a road segment and then we define the performance measure.

The actual travel time of a road segment S is initialized to be the free-flow travel time of S. Whenever a report is produced for S, the actual travel time of S is updated instantaneously, using Equation 2.1 (see §II.C). Intuitively, the actual travel time represents the knowledge that would have been maintained by a perfect dissemination algorithm, i.e., the algorithm that disseminates every report to every vehicle reliably and with no delay. Thus, the difference between the actual travel time and the travel time maintained at local digital maps measures how close a VANET dissemination algorithm is to the perfect case.

Let s_1, s_2, …, s_n be all the road segments in the road network. Let p be the location of a vehicle O at a point t in time. Let $T(s_k)$ be the actual travel time of a road segment s_k at t. Let $T(O,s_k)$ be the travel time of s_k maintained at O's digital map at t. The difference in knowledge of O at t, denoted $DIK(O,t)$, is

$$DIK\ (O,t) = \sum_{k=i}^{n} \left(\frac{1}{g} \mid T(s_k) - T(O,s_k) \mid\right) \qquad (4.1)$$

where g is the free-flow travel time along the shortest-distance path from p to the middle point of s. Intuitively, the difference in knowledge of O is the weighted sum of the difference in knowledge of O between the actual travel times of each road segment and those maintained at the local digital map of O. The farther away the road segment is from O, the lower its weight. Specifically, the weight of a road segment s_k is $1/g$.

The DIK for each vehicle every 10 seconds. The average DIK among all the vehicles throughout the simulation run is taken to be the performance measure.

C. *Simulation Results*

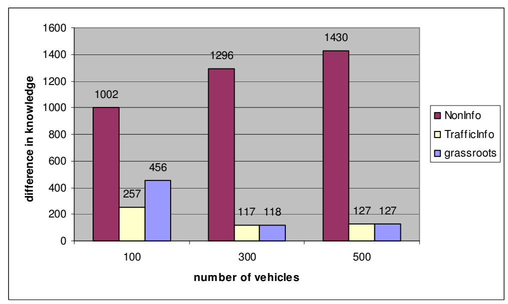

Fig 4.2. Comparison among TrafficInfo, Grassroots, and NonInfo.

Figure 4.2 shows the comparison among the three algorithms. From the figure it can be seen that both TrafficInfo and Grassroots significantly outperforms NonInfo. When the VANET density is high (300 and 500 vehicles), TrafficInfo and Grassroots outperforms NonInfo by an order of magnitude. This quantifies the benefit of VANET dissemination.

Furthermore, when the VANET density is low (100 vehicles), TrafficInfo outperforms Grassroots by 40%. This demonstrates and quantifies the benefit of store-and-forward. More specifically, Grassroots does not store reports and retransmit. In other words, in Grassroots a report can be disseminated only to the vehicles that have a contemporaneous path to the producer when the report is produced. Thus in the area where the network is disconnected, the reports dissemination suffers. In TrafficInfo, via store-and-forward, a report may reach vehicles that are not connected to the producer.

Observe that a low VANET density does not necessarily imply light road traffic. Firstly, around the slow down area, the road traffic remains heavy, even if the overall VANET density throughout the studied road network is low. Secondly, the market penetration, namely the fraction of vehicles participating in VANET dissemination, is not likely to be 100%. When the market penetration is low, the road traffic is much higher than the VANET density. Thus the advantage of TrafficInfo over Grassroots in the low VANET density environment is still important.

Finally, observe that TrafficInfo and Grassroots tie each other when the number of VANET vehicles is 300 and 500. Intuitively, when the VANET density is high, there is no need to do store-and-forward.

V. DISCUSSIONS

The DIK measure. Let us compare DIK with other data dissemination metrics, including throughput, response time, trip-time [4] (the time taken by a vehicle to move from a source to a destination), and resource-discovery-time [8] (the time taken by a vehicle to discover and capture a physical resource such as a parking slot). First, DIK integrates throughput and response time. This is because, in order for the DIK value to be minimized, a vehicle has to receive as many reports and with as small delays as possible. The drawback of the trip-time metric is that the disseminated travel-time information may make vehicles choose common routes so as to avoid a traffic jam. This causes the "herding" effect, which has nothing to do with data dissemination. The DIK metric factors the "herding" effect out. Finally, the DIK metric is general in the sense that it is applicable to various applications such as route-planning and resource discovery.

Incorporation of road type into ranking. As a subject of future work, we plan to incorporate the road type into reports ranking. Intuitively, a report regarding a highway segment is more important than a report regarding a local street segment. A possible way of incorporating the road type into reports ranking is to weight Equation 3.2 by the road type. The weight may be learned in a similar principle by which the supply is learned.

VI. RELEVANT WORK

Message delivery in mobile/vehicular ad-hoc networks. The work in this area is mainly concerned with sending a message to a specific destination given by the network address (see [6] for a survey). In our case the network addresses of the destinations (all the vehicles in the network) are not known a priori. There is a body of work that deals with geographic routing (e.g., [9]) in which a message is routed from a source node to a geographic location or area. However, in most of the existing literature in this area message delivery is possible only if the source and destination are connected, namely there exists a path from the source to the destination.

Disseminating traffic information in VANET's. TrafficView [4] is a store-and-forward approach to disseminating traffic information in VANET's. In TrafficView [4], multiple traffic reports are aggregated into a single report for saving and transmission. Via aggregation the bandwidth consumption is reduced. In our approach, the bandwidth constraint is dealt with via ranking and transmission size control. Thus our approach is orthogonal to and can be combined with TrafficView. The combination of TrafficInfo and TrafficView is a subject of our future work.

VII. CONCLUSION

In this paper we proposed the TrafficInfo algorithm for disseminating real-time traffic information in VANET's. TrafficInfo includes a strategy for a vehicle to prioritize the reports based on their relevance. The relevance of a report depends on its demand (how relevant it is to travel-route planning), and supply (how many vehicles already have it). A machine learning algorithm, called MALENA, is used to enable the estimation of the supply. Furthermore, TrafficInfo adaptively adjusts the number of reports included in a transmission. With such adaptive control of transmission size, the number of collisions is minimized and the available bandwidth is optimally utilized.

We compared the TrafficInfo algorithm with Grassroots, which disseminates reports by simple flooding. The comparison is based on the simulation of vehicle mobility in a real road network and of the 802.11 protocol. The comparison uses a novel performance metric which combines the throughput and response time metrics. TrafficInfo outperforms Grassroots by 40% when the VANET density is small. The two algorithms tie when the VANET density is high.

REFERENCES

[1] Traffic Information for US Cities. http://traffic.com

[2] S. Greaves. Simulating Household Travel Survey Data in Metropolitan Areas, Ph.D. Dissertation, Louisiana State University, 2000.

[3] GCM Travel. http://www.gcmtravel.com

[4] S. Dashtinezhad, T. Nadeem, C. Liao, and L. Iftode. Trafficview: A scalable traffic monitoring system. *MDM*, 2004.

[5] S. Goel, T. Imielinski, K. Ozbay. Ascertaining viability of WiFi based vehicle-to-vehicle network for traffic information dissemination. The 7th *Int. Conf. on Intelligent Transportation Systems*, 2004.

[6] E. Royer and C. Toh, A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks, *IEEE Personal Communications*, pages 46-55, Apr. 1999.

[7] R. Barr. An efficient, unifying approach to simulation using virtual machines. PhD thesis, Cornell University, 2004.

[8] O. Wolfson, B. Xu, and H. Yin. Reducing Resource Discovery Time by Spatio-Temporal Information in Vehicular Ad-Hoc Networks. *VANET*, 2005.

[9] T. Imielinski, J. Navas. GPS-Based Geographic Addressing, Routing, and Resource Discovery. *Communications of the ACM*, 42(4), pp 86-92, 1999.

[10] O. Wolfson, B. Xu, H. Yin, H. Cao. Search-and-Discover in Mobile P2P Network Databases. *ICDCS*, 2006.

[11] Y. Huang and H. Garcia-Molina. Publish/Subscribe Tree Construction in Wireless Ad-Hoc Networks. *MDM*, 2003.

[12] O. Wolfson and B. Xu. Mobile Peer-to-peer Data Dissemination with Resource Constraints. *MDM'07*.

[13] O. Wolfson and B. Xu. Flooding by Machine Learning in Mobile Ad Hoc Networks. Accepted to appear in *ACM Trans. on Autonomous and Adaptive Systems* (TAAS) on Autonomic Communication.

[14] David R. Choffnes and Fabián E. Bustamante. An Integrated Mobility and Traffic Model for Vehicular Wireless Networks. *VANET* 2005.