OPERA: Opportunistic Packet Relaying in Disconnected Vehicular Ad Hoc Networks

Mahmoud Abuelela Stephan Olariu
Department of Computer Science
Old Dominion University
Norfolk, VA 23529-0162

Ivan Stojmenović
SITE, University of Ottawa
Ottawa, Ontario, K1N 6N5
CANADA

***Abstract*—Vehicular Ad Hoc Networks (VANET) have recently received increasing attention in the media. And with good reason: VANET promises to integrate driving into a ubiquitous and pervasive network that will redefine the way we live and work.**

Empirical evidence, accumulated over time, has shown that under many highway scenarios VANETs tend to be disconnected, consisting of a collection of disjoint clusters. Our first contribution is to provide an analytical explanation of this rather counterintuitive result, thus confirming the findings of [2], [7], [8], [15]. We show that this phenomenon is present even in relatively dense traffic and provide an analytical expression of the expected size of a cluster, as a function of traffic density and communication range. We also show that the cluster size is quite stable and easy to maintain.

In disconnected VANET, packet relaying takes on a new meaning: a vehicle that stores a packet may have to carry it for a while, acting as a "data mule", before a suitable next hop can be identified. Although some protocols have been introduced to take advantage of this fact, they are still suffering from some disadvantages such as wasting bandwidth and delayed propagation. Motivated by this fact, our second contribution is an Opportunistic Packet Relaying protocol (OPERA) for disconnected VANETs. In OPERA a packet progresses towards destination opportunistically, by a combination of data muling and local routing with the help of both co-directional and oncoming clusters in a smart way to avoid wasting resources. Finally, we offer simulation results to evaluate the performance of OPERA.

I. INTRODUCTION AND MOTIVATION

In the US, VANETs are using 75MHz of spectrum in the 5.850 to 5.925 GHz band specially allocated by the US Federal Communications Commission (FCC) for Dedicated Short Range Communications (DSRC) [4]. While the original impetus for VANET was provided by traffic safety [4], more recent concerns involve privacy and security as well as integrating VANET into the fast-growing *infotainment* industry [10]. The potential societal impact of VANET is confirmed by the proliferation of consortia and initiatives involving car manufacturers, government agencies and academia. Examples include, among others, the Car-2-Car Communication Consortium, the Vehicle Infrastructure Integration Program, the Vehicle Safety Consortium, and the Advanced Safety Vehicle Program [3], [4].

In spite of their close resemblance to MANET, with which they share the same underlying philosophy, VANET networks have a number of specific characteristics that set them apart from MANET. First, while most MANET networks are deployed in support of special-purpose operations including disaster relief, search-and-rescue, law-enforcement and multimedia classrooms, all of which are intrinsically short-lived and involve a small number of nodes, VANET networks may involve thousands of fast-moving vehicles over tens of miles of roadways and streets. Second, and perhaps more importantly, while MANET networks may experience *transient* periods of loss of connectivity, in VANET, especially under sparse traffic conditions, extended periods of disconnection are the norm rather than the exception. This state of affairs has a significant implication on routing, rendering traditional MANET routing protocols unsuitable for VANET. To address the specific needs of VANET, several routing protocols have been proposed.

A. Our contributions

To set the stage for describing our contributions, imagine a car cruising down a highway with possibly multiple lanes of traffic in each direction. It is clear that, per time unit, our car will meet far more oncoming cars than co-directional vehicles. This is due to the fact that the *relative speed* of cars moving in opposite directions is the sum of their absolute speeds, while co-directional cars appear to be moving, relative to one another, at the difference of their absolute speeds. In particular, if the majority of co-directional cars are keeping the legal speed, the relative speed of our car with respect to the traffic going in the same direction is almost nil. As a result, routes established between co-directional cars tend to be *stable*. This observation has motivated a number of workers to propose establishing routes consisting entirely of co-directional cars. However, it was recently noticed [1], [2], [7], [15] that in many highway scenarios co-directional traffic consists of disjoint clusters with no

1-4244-2575-4/08/$20.00

connectivity between them. This, in turn, implies that end-to-end connectivity between co-directional cars is not guaranteed to exist.

Indeed, while acknowledging the existence of clusters and providing an approximation of the expected inter-cluster distance, Agarwal *et al.* [2] does not explain why co-directional traffic is inherently disconnected. Our first contribution is to provide an analytical explanation of this rather counterintuitive phenomenon. We also show that clustering is present even in relatively dense traffic and provide an analytical expression of the expected size of a cluster showing that clusters are stable over some period of time.

To motivate our second contribution, imagine that car *D* wishes to receive some information such as traffic conditions, a popular movie, or the transmission of an ongoing live sporting event. For this purpose it floods a request, together with its position and speed information, to the cars *ahead* of it (both co-directional and oncoming). Assume, without loss of generality, that some car *S* in the oncoming direction has the desired packet(s) for *D*. In this paper we focus on the *delivery phase* of this scenario, namely routing the packet(s) from *S* to *D*. Because of the small relative speed of cars in a cluster, the cluster structure is assumed to remain stable and easy to maintain. In this context, our second contribution is to develop an Opportunistic Packet Relaying protocol (OPERA) for disconnected VANET, for solving the stated problem. OPERA takes advantage of the relative stability of co-directional clusters to maintain proactively routing information in each co-directional cluster as well as information about overlapping oncoming clusters. Packet relaying is achieved, opportunistically, by a combination of data muling and local routing with the help of oncoming clusters in a clever way to avoid possible delays and bandwidth wasting. We prove that OPERA is time-optimal in the sense that no protocol can deliver a packet faster. Extensive simulation results have revealed that OPERA outperforms DPP [7] in terms of packet delivery time and resource utilization.

The paper is organized as follows: Section II reviews relevant work; Section III present analytical results concerning expected inter-car separation and the probability of gaps in co-directional traffic. Section IV presents a detailed discussion of OPERA, followed, in Section V by performance evaluation. Finally, Section VI offers concluding remarks.

II. RELATED WORK

GVGrid [19] is an on-demand, position-based routing protocol that constructs a route from a static source node to vehicles that may exist in a destination region. It also maintains the route when it breaks because of vehicle mobility. GVGrid tries to discover, based on vehicle mobility characteristics, a route that is expected to provide the best stability. Unfortunately, in disconnected VANET GVGrid may not work as intended.

MURU [16] is a multi-hop routing protocol intended to find robust paths in urban VANET. MURU tries to minimize the probability of path breakage by exploiting mobility information of each vehicle and by using a special parameter called expected disconnection degree factor to select the most robust path from source to destination. MURU implicitly assumes that there will be many paths between source and destination and it strives to select the most stable one. However, this assumption is incorrect in highway scenarios when traffic decomposes into disconnected clusters.

In [5] an interesting Position Based Routing (PBR) protocol is discussed where packet forwarding decisions are made based on power awareness. The basic routing strategy is a variant of greedy forwarding where the next hop is selected to be the car closest to the destination. While this strategy is correct, it may lead to unnecessary forwarding and, ultimately, to wasted bandwidth.

AMB [20] is an interesting infrastructure-independent multi-hop broadcast protocol. In AMB, the furthest node in the broadcast direction is assigned the function of forwarding and acknowledging the packet. AMB selects the closest car to the intersection to act as a repeater that broadcasts the packet to all other directions. AMB differs from OPERA in that they focus on an urban environment where end-to-end connectivity between intersections is assumed.

Connectivity-Aware Routing (CAR) [1] was developed taking into consideration the fact that end to end connectivity is not guaranteed in VANET. The main idea of CAR is to try finding a connected path between the source and destination while applying the idea of data mules or carry and wait when there exits some form of temporal disconnectivity.

Directional Propagation Protocol (DPP) [7], [8] utilizes the directionality of data and vehicles for packet propagation. DPP considers real traffic scenarios in which vehicles form clusters on the road and these clusters may be disconnected from each other. It uses co-directional clusters, that is, clusters that run in the same direction as the packet to deliver. When disconnection occurs between two co-directional clusters, clusters in the opposite direction are used as bridges to the next co-directional cluster. However, as we show in the paper, DPP is likely to waste significant bandwidth because of uneven traffic density as well as imposing delays on packets propagation. Worse yet, in Subsection IV-B we show that under certain conditions DPP is not guaranteed to deliver a packet at all.

In [2], an analytical model for DPP is introduced in which the expected distance between clusters, the expected disconnection time and the effective propagation rate were computed. However, the model of [2] does not explain why traffic clustering is inherent to VANET. By contrast, we offer an analytical models for VANET that allows us to compute the probability of end-to-end connectivity under different traffic density. Our model shows that end-to-end connectivity may not be present even in relatively dense traffic. We also obtain the expected size of a cluster.

In [9], a realistic traffic scenario is considered in which cars may form clusters that are disconnected from each other. A hybrid routing protocol is introduced that can route a packet inside a cluster but relies on a pre-existing infrastructure to connect these clusters. Although, this protocol considers real traffic situations, its reliance on a pre-existing infrastructure is problematic. Indeed, the cost of installing infrastructures along roadway is prohibitively expensive.

Among others like GyTAR [21], GPCR [22] and many others that we could not describe because of stringent page limitations.

III. EVALUATING THE PROBABILITY OF LARGE GAPS IN CO-DIRECTIONAL TRAFFIC

While traffic displays diverse spatio-temporal patterns, several workers have pointed out that an *instantaneous* snapshot of a steady free flow of uncongested traffic can be approximated by uniform car density (measured in cars per kilometer), which translates into a uniform vehicular distribution [13], [17], [18].

The goal of this section is to provide an answer to the following natural question: Given that m cars are deployed uniformly at random in a single lane of traffic of one kilometer and given that dependable radio communications between cars require a *maximum* inter-car distance of 200 meters, what is the probability that there is end-to-end radio connectivity between the m cars? This question is fundamental. We prove that, surprisingly, the number of cars per kilometer must be at least 16 in order to have a better than even chance for connectivity; it takes about 25 cars per kilometer for end-to-end connectivity to be present with 90% probability.

A. The analytical model

Returning to our problem, we model the situation as follows: the m cars determine $m-1$ *distinguishable* bins (inter-car spaces), enumerated in left-to-right order as $B_1, B_2, \dots B_{m-1}$. The number of distinguishable ways in which the n indistinguishable balls (unit inter-car spaces) can be distributed into the $m-1$ bins is easily seen to be $\binom{m+n-2}{n} = \binom{m+n-2}{m-2}$. To see that this is the case, observe that the $m-1$ bins involve m separators and that we can lay down the balls and bins in a linear sequence flanked on both sides by a separator. The problem now is that of selecting n places for the balls out of a total of $n+m-2$ places available. The conclusion follows.

Now suppose that we want a *given* bin to contain k, $(0 \leq k \leq n)$, balls. This amounts to distributing k balls into one bin and $n-k$ balls into the remaining $m-2$ bins. Reasoning as above, the number of distinguishable ways in which this can be achieved is $\binom{(n-k)+(m-3)}{n-k} = \binom{n+m-k-3}{n-k}$. As a consequence, the probability p_k, $(0 \leq k \leq n)$, of the event that a given bin contains *exactly* k balls is

$$p_k = \binom{n+m-k-3}{n-k}\binom{m+n-2}{n}^{-1}. \qquad (1)$$

To show that the p_ks are a valid probability distribution, we need to prove that $\sum_{k=0}^{n} p_k = 1$. This, in turn, amounts to showing that $\sum_{k=0}^{n} \binom{(n-k)+(m-3)}{n-k} = \binom{m+n-2}{n}$. Indeed, recalling that for integers r and n,

$$\sum_{t \leq n} \binom{r+t}{t} = \binom{r+n+1}{n} \qquad (2)$$

(see [6], (5.9) p.159), we write

$$\begin{aligned}
\sum_{k=0}^{n} p_k &= \binom{m+n-2}{n}^{-1} \sum_{k=0}^{n} \binom{(n-k)+(m-3)}{n-k} \\
&= \binom{m+n-2}{n}^{-1} \sum_{i \leq n} \binom{(m-3)+i}{i} \\
&= \binom{m+n-2}{n}^{-1} \binom{(m-3)+n+1}{n} \text{ [by (2)]} \\
&= \binom{m+n-2}{n}^{-1} \binom{m+n-2}{n} \\
&= 1,
\end{aligned}$$

as desired.

In our setup, two neighboring cars become disconnected if the bin corresponding to the distance between them accumulates at least $d+1$ balls, where d corresponds to the maximum effective transmission range. Let A_i, $(1 \leq i \leq m-1)$, be the probability that a generic bin B_i contains *at least* $d+1$ balls.

Lemma 3.1: For all i, $(1 \leq i \leq m-1)$,

$$\Pr[A_i] = \binom{m+n-(d+1)-2}{m-2}\binom{m+n-2}{n}^{-1}$$

.

Proof: We find it convenient to compute the probability of the complementary event $\overline{A}_i$. By (1) and (2) we can write

$$\begin{aligned}
\Pr[\overline{A}_i] &= \binom{m+n-2}{n}^{-1} \sum_{j=0}^{d} \binom{(m-3)+(n-j)}{n-j} \\
&= \binom{m+n-2}{n}^{-1} \sum_{t=n-d}^{n} \binom{(m-3)+t}{t} \\
&= \binom{m+n-2}{n}^{-1} \sum_{t=0}^{n} \binom{(m-3)+t}{t} \\
&- \binom{m+n-2}{n}^{-1} \sum_{t=0}^{n-d-1} \binom{(m-3)+t}{t} \\
&= 1 - \binom{m+n-2}{n}^{-1} \binom{m+n-d-3}{m-2}
\end{aligned}$$

Thus, $\Pr[A_i] = 1 - \Pr[\overline{A}_i] = \binom{m+n-2}{n}^{-1}\binom{m+n-d-3}{m-2}$, and the proof of the lemma is complete. ■

Let A be the event that there is *no* end-to-end connectivity between the m cars. Clearly, $A = \cup_{i=1}^{m-1} A_i$. Since the A_i's are not independent, the principle of *inclusion-exclusion* implies that $\Pr[A] = \sum_{i=1}^{m-1} Pr[A_i] - \sum_{1\le i<j\le m-1} \Pr[A_i \cap A_j] + \cdots + (-1)^i \sum_{1\le j_1<j_2<\ldots<j_i\le m-1} \Pr[A_{j_1} \cap A_{j_2} \cap \cdots \cap A_{j_i}] + \cdots$

Lemma 3.2: For all i, j, $(1 \le i < j \le m-1)$, $\sum_{1\le i<j\le m-1} \Pr[A_i \cap A_j] = \binom{m-1}{2}\binom{m+n-2(d+1)-2}{m-2}\binom{m+n-2}{n}^{-1}$.

Proof: We provide a purely combinatorial proof. First, to obtain $\Pr[A_i \cap A_j]$, observe that the number of distinguishable arrangements in which bins i and j contain at least $d+1$ balls is obtained by first placing $d+1$ balls in bins i and j and then by distributing the remaining $n-2(d+1)$ balls uniformly at random in *all* the $m-1$ bins. This can be done in $\binom{m+n-2(d+1)-2}{n-(d+1)} = \binom{m+n-2(d+1)-2}{m-2}$ distinct ways. Since there are $\binom{m-1}{2}$ distinct ways of choosing i and j subject to $(1 \le i < j \le m-1)$, the conclusion follows. ■

Lemma 3.3: For all $1 \le j_1 < j_2 < \ldots < j_i \le m-1$, $\sum_{1\le j_1<j_2<\ldots<j_i\le m-1} \Pr[A_{j_1} \cap A_{j_2} \cap \cdots \cap A_{j_i}] = \binom{m-1}{i}\binom{m+n-i(d+1)-2}{m-2}\binom{m+n-2}{n}^{-1}$.

Proof: Follows from Lemma 3.2 by a simple inductive argument. Details omitted because of space limitation. ■

Theorem 3.4:

$$\Pr[A] = \frac{\sum_{i=1}^{m-1}(-1)^{i+1}\binom{m-1}{i}\binom{m+n-i(d+1)-2}{m-2}}{\binom{m+n-2}{n}}. \quad (3)$$

Proof: Follows directly from Lemmas 3.1, 3.2 and 3.3, combined. ■

Although a closed form for $\Pr[A]$ is hard to obtain, we have compared the results obtained by evaluating (3) for various values of m with those yielded by averaging 10 million simulations of an experiment that consists in generating uniformly at random m points in the unit interval and checking whether any two neighbors are separated by more than 0.2. As illustrated in Figure 1, our simulation results are virtually indistinguishable from the analytical result.

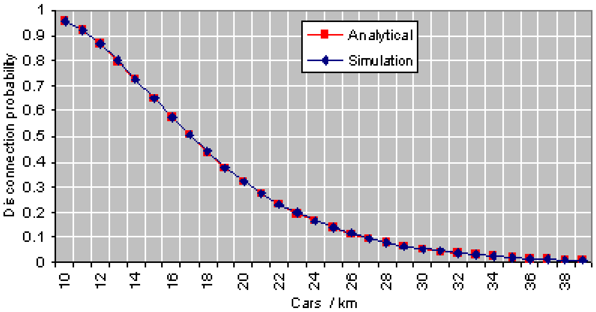


Fig. 1. *Illustrating co-directional disconnection probability.*

One can interpret (3) as follows. Imagine sliding a 1 Km window down a highway with one lane of traffic in each direction. If the window contains m co-directional cars, then the probability that there is no end-to-end connectivity between them is precisely $\Pr[A]$ in (3). For example, should there be 12 co-directional cars in the window, the probability of no end-to-end connectivity between them is about 86%. Naturally, the probability *decreases* with the number of co-directional lanes of traffic in each direction.

B. Evaluating the expected size of a cluster

Since, as we saw, co-directional traffic is *inherently* partitioned into clusters, an interesting question is to estimate the expected size of a cluster. The goal of this subsection is to provide an answer to this natural question. For this purpose, we inherit the notation and terminology of Subsection III-A.

Theorem 3.5: The expected size of a cluster is

$$E[cluster\_size] = \frac{m \cdot \binom{m+n-2}{n}}{\binom{m+n-2}{n} + (m-1)\cdot\binom{m+n-d-3}{n}}. \quad (4)$$

Proof: As we saw, the probability p that a given bin contains at *least* $d+1$ balls is $p = \binom{m+n-d-3}{m-2}\binom{m+n-2}{n}^{-1}$. Let X be the random variable that counts the number of "gaps" (i.e., the number of bins containing at least $d+1$ balls). Since X is binomial, the expected value $E[X]$ of X is

$$\begin{aligned}
E[X] &= (m-1)\cdot p \\
&= (m-1)\cdot\binom{m+n-d-3}{m-2}\binom{m+n-2}{n}^{-1} \quad (5)
\end{aligned}$$

Once we have the expected number of gaps in co-directional traffic, the expected number of clusters becomes $1 + E[X] = 1 + (m-1)\cdot\binom{m+n-d-3}{m-2}\binom{m+n-2}{n}^{-1}$.

Thus, the expected size of a cluster is

$$E[cluster\_size] = \frac{m \cdot \binom{m+n-2}{n}}{\binom{m+n-2}{n} + (m-1) \cdot \binom{m+n-d-3}{n}},$$

completing the proof of the theorem. ■

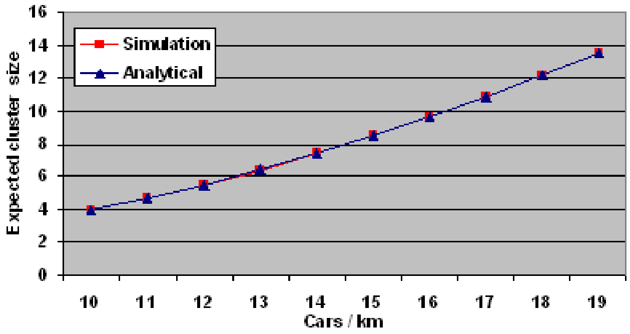


Fig. 2. *Illustrating the expected cluster size.*

Figure 2 provides a side-by-side comparison of the expected cluster size predicted by (4) and the value obtained by simulation. As an illustration, imagine a two-lane road of 1Km and 10 cars distributed uniformly at random per lane of traffic. By virtue of (5) we expect to see about 2.47 clusters; by (4), we expect a cluster to contain between 4 and 5 cars.

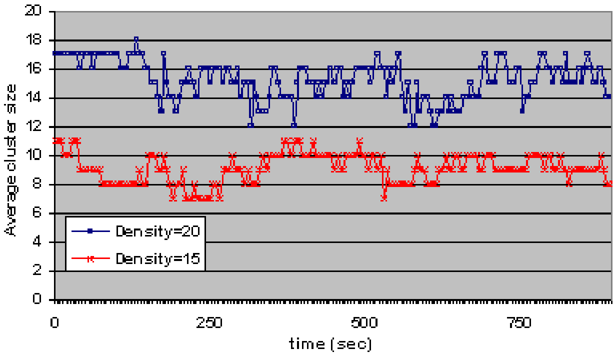


Fig. 3. *Illustrating cluster stability.*

C. Cluster stability

As already mentioned, since co-directional vehicles move at a small relative speed with respect to each other, we expect clusters to be quite stable and easy to maintain. We defer discussing cluster maintenance until Subsection IV-D. In order to get a better understanding of co-directional cluster dynamics we have simulated a stretch of highway with two free traffic flow of 15 and 20 cars/km. In both cases, the difference between the highest and lowest speed is 15km/hour. Figure 3 illustrates, side by side, the average cluster sizes over 15 minutes of simulation time. In both cases the simulation revealed that in spite of mobility, the expected cluster size is remarkable close to the theoretical prediction of 10 and 15 cars, respectively. Incidentally, this is also indirect validation of the uniformity assumption.

IV. OPERA: OUR OPPORTUNISTIC PACKET RELAYING PROTOCOL

A. The vehicle and communication model

We assume cars to be GPS-enabled and to communicate using DSRC [4]. Being GPS-enabled, the cars know their geographic position and are synchronized. As mandated by DSRC, every 300 ms each vehicle sends a beacon with a range of about 200–300 m. This beacon contains information that allows vehicles to handshake and synchronize. OPERA uses these beacons for cluster formation and cluster maintenance as well. Mindful of their original intent we shall, nonetheless, refer to these beacons as *Cluster Management Beacons*, (CMB, for short). So, it is noteworthy here to mention that OPERA has almost *zero* overhead in maintaining these clusters as it uses these beacons that are transmitted anyway for many other purposes. We refer the reader to Figure 4 for the proposed format of a CMB obtained by taking advantage of *unused* fields of the standard IEEE 802.11 beacon used by DSRC. As in IEEE 802.11, the length of the timestamp field is 8 bytes, the length of the beacon interval is 2 bytes, typical value is 300 ms, the GPS position may be encoded in 12 bytes [12]; each speed field may be encoded in 2 bytes. Finally, there is a 4-byte unused filed. Thus, the CMB beacon size is 56 bytes.

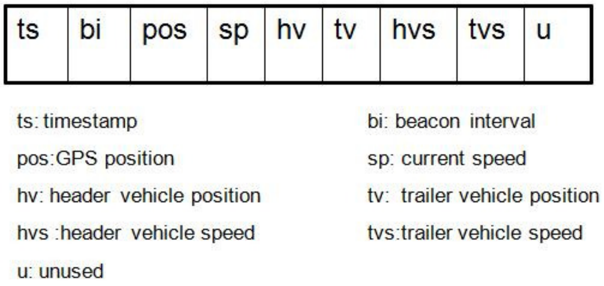


Fig. 4. *Illustrating the layout of a CMB.*

B. A motivating example

Referring to Figure 5, suppose that car a wishes to deliver a packet to car g. The simplest and most straightforward strategy, that does not require routing at all, is for car a to wait until, eventually, it meets car g and can deliver the packet directly, as illustrated in Figure 5(c). In such a scenario we say that car a acts as a *data mule* for car g [11]. Muling guarantees delivery and is quite effective if the geographic distance between the two cars is modest and traffic is sparse. In very

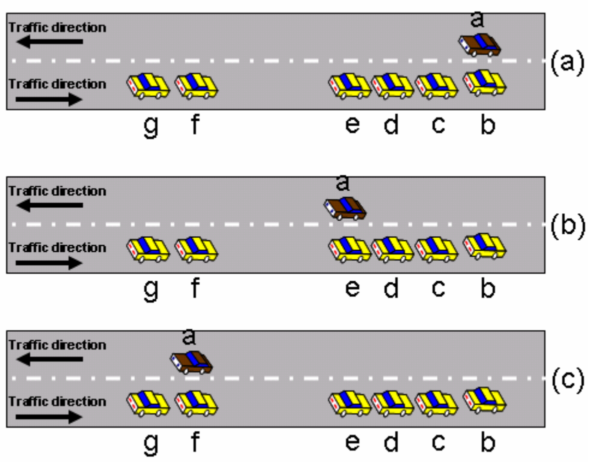


Fig. 5. *OPERA – a motivating example.*

sparse traffic muling may be the only workable strategy. In some cases, however, one can do better than muling.

The majority of zero-infrastructure VANET routing protocols assume end-to-end connectivity of co-directional traffic with some exceptions like DPP [7] and CAR [1] that observed that co-directional cars are grouped into clusters. These protocols would work as follow. Consider the example in Figure 5(a). Car a detects that the only vehicle in range is car b and sends the packet to it. Since the cars b,c,d,e form a cluster, the packet sent to car b, will be multi-hopped, in the obvious way, to car e. Car e will keep the packet for a while until it meets car a to which it will upload the packet as shown in Figure 5(b). Thus, the packet that originated at car a ends up at car a, again. Clearly, such a situation is most undesirable since a sizable amount of resources has been consumed (in signaling and routing the packet) and has achieved nothing. In this case valuable bandwidth was wasted in routing from a to e (along the chain b, c, d, e) and then back to a. It would have been much better for car a not to send the packet to b at all.

In addition to wasting bandwidth, most of the existing protocols are suffering from many other disadvantages like routing loops that may exist in the previous example if car e misses the connection with car a, it may send the packet to any car behind a that would in turn be routed again to a!.

Actually, we can show several examples to illustrate that DPP, and other existing zero-infrastructure protocols, are either not efficient or may not deliver the packet correctly.

In OPERA , a packet may "hop" between clusters or cars moving in opposite lanes until, eventually, it reaches its destination. In this sense, OPERA is actually a *hybrid* protocol as it alternates between applying proactive routing and data muling in a clever way to avoid delays or bandwidth wasting.

C. OPERA – Cluster formation

To begin, the task of clustering can be performed as follows: each car that has not received, within a certain time-out interval, a beacon from a co-directional car in front of it, declares itself leader of the cluster and sends this information in the next CMB to the cars behind it. The message will be then multi-hopped, using CMB beacons throughout the cluster. Note that this information is piggybacked in regular beacons transmitted by cars. Thus, OPERA has almost zero overhead over the slandered beacons. The CMB contains, in addition to the identity of the leader, its geographic position, direction of movement and speed. (Suffice it to say that direction is immediately available if speed if kept as a signed integer.) Every co-directional car that receives such a CMB understands that it belongs to the cluster named after the leader. In a symmetric way, the last car in the cluster informs, by virtue of a CMB, all the other cars in its own cluster of its geographic position and speed. The *head* and *tail* vehicles in a cluster, maintained proactively as described, play a special role in OPERA and will be denoted, respectively, by $h(\cdot)$ and $t(\cdot)$. Since co-directional vehicles move at a small relative speed with respect to each other, we expect clusters to be quite *stable* and easy to maintain.

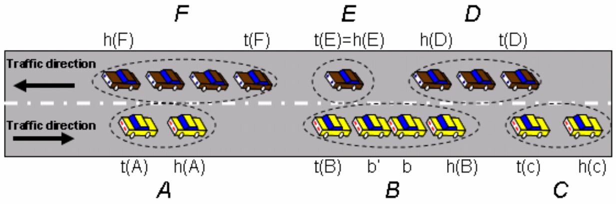


Fig. 6. *Illustrating clusters on a two-lane highway.*

D. OPERA – Maintaining cluster-related information

A generic vehicle x in a cluster X maintains *proactively* information about the cluster to which it belongs as well as *overlapping* clusters, that is, clusters in the oncoming direction with which some node in x's cluster is in direct radio contact. By propagating this information during cluster formation and maintenance, every car in the cluster acquires information that allows them to make adequate routing decisions. This ability to make routing decisions is of key importance in OPERA. We note that all the bindings described below are soft and are subject to time-out. Specifically, x maintains:

- A first record $(x.cid, x.pos, x.sp, x.ns, x.ncids)$ where
  - $x.cid$ is the identity of the cluster to which x belongs,
  - $x.pos$ is the current position of x obtained from the on-board GPS device,

  - $x.sp$, is the current speed obtained from the on-board speedometer,
  - $x.X$, the sets of all neighboring cars in x's cluster that can be reached from x in one hop, these can be easily maintained from their beacons.
  - $x.O$, the sets of all neighboring cars in overlapping clusters that can be reached from x in one hop.
- A second record, also maintained proactively as described in subsection IV-C, contains information about head and tails of $x.X$.
- A third record contains a flag to determine weather there is any overlapping cluster in advance or not and the expected time to loose such overlap. This information may be maintained as follow. Once car y detects that $y.O$ is not empty, it can piggyback such information in its next CMB beacon. By exchanging beacons, all cars in the $y.Y$ will be aware about such overlap and its expected duration.

It is important to realize that, by virtue of Theorem 3.5, the cluster size is bounded (see also Figure 2). Moreover, since co-directional clusters tend to be *stable*, and the underling topology of clusters *linear*, maintaining these records proactively is not a problem and we do not run into scalability problems.

Moreover, we will show shortly that these beacons introduce a small overhead in terms of bandwidth wasting while on the other hand they save much bandwidth and time in packets propagation. As clusters in opposite directions "meet", they exchange routing information. This allows the cars in each cluster to update their routing tables. Since the bindings are soft, as these clusters drift away from each others, the information is no longer reinforced and will be removed.

E. OPERA – the details

With these preliminaries out of the way, the next two sections are devoted to the details of OPERA. We distinguish between the *Baseline Algorithm*, discussed in Subsection IV-F, and the *General Algorithm* discussed in Subsection IV-G.

F. OPERA – the Baseline Algorithm

The metric we adopt for assessing the performance of OPERA is *delivery time* and, for the same delivery time, *hop count*. The main reason for this choice is that pure *data muling* achieves optimal hop-count, in fact, a hop count of 1, (see Figure 5(c)), at the expense of delivery time. An immediate corollary of this observation is that in order to optimize delivery time, packets that cannot be routed to an overlapping cluster will be routed, internally, to $h(.)$, the first car in the cluster. Nonetheless, to give the reader the full generality of the situation, in the Baseline Algorithm discussed below, we assume that an overlapping cluster is available and that the packet to be routed is stored by an arbitrary car, not necessarily the first car in the cluster.

The *Baseline Algorithm* that we discuss in this subsection is the workhorse of OPERA. $Baseline(a,A,x,X)$ assumes that some vehicle a in cluster A has a packet to relay to a vehicle x in cluster X such that the following conditions are satisfied:

- a and x are in opposite lanes of traffic, and
- $A \cup X$ is a connected graph.

Refer to Figure 7 for an illustration.

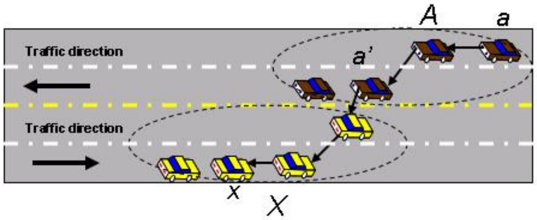


Fig. 7. *Illustrating the Baseline Algorithm.*

Two nodes are *within range* of each other if their radio connection lasts longer than packet transmission time. Since the cars know their location, this is easy to determine. We assume, without loss of generality, that x is to the left of a.

$Baseline(a,A,x,X)$ works as follows. If x is within range of a, the packet is delivered directly. Otherwise, by consulting its routing table, list of all 1-hop neighbors, a forwards the packet to a car (in either A or X) that minimizes the hop-count to x and start the baseline again.

Lemma 4.1: Assuming correct cluster-related information, $Baseline(a,A,x,X)$ correctly relays the packet from a to x along a shortest path in $A \cup X$.

Proof: The correctness and the optimality of the baseline Algorithm follow directly by the choice of the next hop, one that minimized the hop-count to x. ■

G. OPERA – the General Algorithm

The General Algorithm, $General(a,A,x,X)$, assumes that vehicle a in cluster A has a packet to relay to some oncoming vehicle x in (known) cluster X but that the graph $A \cup X$ is not connected. In the terminology of Subsection IV-D, let B be the closest co-directional cluster to X that overlaps with A. Further, let D, if any, be the leftmost cluster among the clusters that overlap with B. We refer the reader to Figure 8 for an illustration. Specifically, in Figure 8(a) cluster B overlaps only cluster A and so, $A = D$; in Figure 8 (b) cluster B overlaps three clusters, namely D, C and A, in left-to-right order.

Let S be the graph induced by A, B and all the clusters overlapping B. It is clear that S is connected and, therefore, one can route the packet held by car a

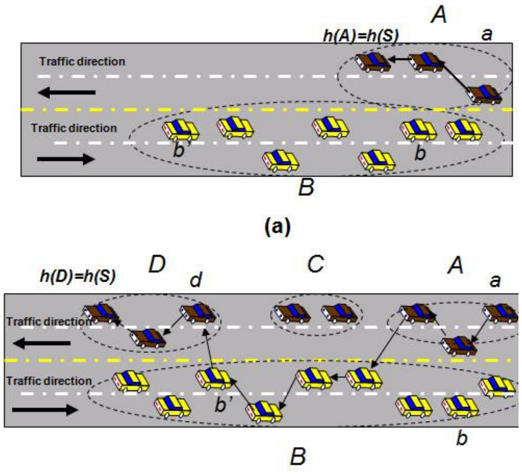


Fig. 8. *Illustrating the General Algorithm.*

to the head of the leftmost cluster co-directional with A in S. By abusing notation a little, we let $h(S)$ denote the head of the leftmost cluster that overlaps B. Referring to Figure 8, $h(S)$ is either the head of cluster A in Figure 8 (a) or the head of cluster D in Figure 8(b). This routing decision is justified by our motivating example and the discussion in Subsection IV-F.

The routing itself can be performed by the following greedy approach. First, if $h(S) = h(A)$ then, clearly, all that needs to be done is to route the packet to the head of a's cluster. If, however, $h(S) = h(D)$, for the leftmost cluster D co-directional and overlapping B, then the packet is routed using the Baseline Algorithm to one of cars, say, b' that has connectivity to some car in cluster D and then by the Baseline Algorithm again, to $h(S)$. The algorithm now continues recursively, until the packet is delivered to x. We emphasize that car a does not know and actually need not to know about clusters C and D: all it knows is that cluster B overlaps with some clusters ahead of its own. So, *a formaly* does the following. *If it realizes that the oncoming cluster has some overlaping with another co-directional cluster on the road, a can know that from beacons broadcasted by 1-hop cars in B, then the baseline algorithm is applied until the packet reaches that next cluster. Otherwise, it routes the packet to the header car.* Note that the above algorithm is repeated at each node until the packet reaches its destination.

Theorem 4.2: Assuming correct cluster-related information and no packets loose, OPERA correctly relays the packet from a to d guaranteeing optimal delivery time.

Proof: We inherit the notation from the previous discussion and proceed by induction on the number, k, of clusters co-directional with X that lie *strictly* between the cluster X (containing x) and A. Also, for simplicity we assume that a is the head of A.

To settle the basis of the induction, note that if $k = 0$ and A and X overlap, then the packet is routed using the baseline algorithm; if, on the other hand, A and X do not overlap, then the packet will be carried by a until A and X overlap at which point it will be delivered by the baseline algorithm. By Lemma 4.1 this is time-optimal.

Next, let k, $(k \geq 1)$, be arbitrary and assume the statement true for all $0 \leq t < k$. In this case, car a (assumed to be the head of cluster A) carries the packet until cluster A overlaps the first cluster, say B, co-directional with X. Since $k > 0$, B and X are distinct. Now, one of the two scenarios in Figure 8 must occur. In either case, the packet will be relayed to $h(S)$ in optimal time. At this point the number of clusters co-directional with X that lie *strictly* between the cluster X (containing x) and A has decreased by one and, by the induction hypothesis, OPERA relays the packet to x time-optimally. This completes the proof of the theorem. ■

V. PERFORMANCE ANALYSIS

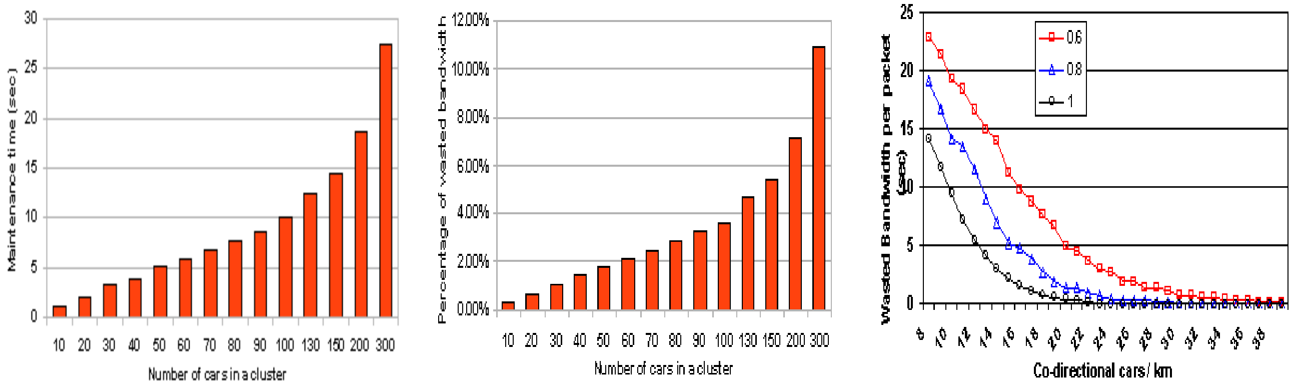
Section II showed that some protocols that use the idea of data mules exist. However, all of the current existing protocols apply the idea of data mules only when a disconnection occurs. One of the key strengths of OPERA is to mix the idea of data mules and routing in *a smart way* that saves both bandwidth and time. In OPERA even if a local route seemed to exist, it may be better for the current car to carry the packet until it finds better route that guarantee fast and efficient delivery based on some performance metrics. Unlike DPP, any intermediate car in OPERA, not only the head and tail, can decide the current optimal route based on the most recent information.

A. Simulation model

We simulated a 4 km stretch of highway, with two lanes of traffic in each direction. Vehicles are deployed at random then each vehicle picked a speed from the interval between 65 mph and 75 mph. As cars move, they may accelerate when possible or decelerate to maintain some safety parameters such as cars inter-spacing. The above mobility model is implemented over 802.11b protocol after modifying it to send beacons periodically every 300 ms. If a car detects a collision, received a corrupted beacon, it may choose another random time to start from. As in [17], the size of vehicles is ignored.

B. Simulation results and analysis

1) Clustering and messages overhead: Figure 9(a) shows the impact of cluster size on the maintenance time, time until all cars in a cluster know about its head and tail. As shown in the figure, for a relatively large cluster size of 60 cars, that may cover a distance over kilo meter,



(a) (b) (c)

Fig. 9. *Clustering overhead per cluster and DPP overhead per packet*

only 5 sec is needed for all cars to learn about head/tail and all 1-hop neighbors. Joining this result with those obtained in figures 2 and 3, cluster size usually is much less than 60, we can claim that cars will have up to the second information about their clusters most of the time.

A natural question that may arise now is how much bandwidth is being wasted in the clustering process? In spite of being sent any way for general purposes like handshaking and synchronization, we measured the percentage of wasted bandwidth in beacons broadcasting. Figure 9(b) shows the impact of cluster size on the wasted bandwidth during maintenance. As shown in the figure, a very small amount of bandwidth is being wasted. For example, up to 60 cars in a cluster waste only about 2% of the bandwidth. Even in large clusters of 300 cars that would cover a distance more than 15 km, assuming 50 m interspacing, only 11% of the bandwidth would be wasted. Thus, clustering in most cases is a lightweight process that consumes a little resources. Actually, we can claim that in almost all cases, cars would have up to the *minute* information about its cluster. Moreover, in a very congested traffic, any greedy packet forwarding would do the job. It is obvious here that we did not restrict ourselves to the 4 km stretch of road mentioned before.

Figure9(c) shows bandwidth wasted by DPP, number of extra messages sent by DPP over OPERA, when oncoming traffic density is 100%, 80% and 60% of the co-directional direction density. As shown in the figure, DPP has a significantly larger overhead than OPERA. As we showed before in Subsection 5, DPP sends unnecessary messages when the co-directional cluster has no overlap with two adjacent oncoming clusters. Hence, co-directional cluster will not be able to work as bridge between two oncoming clusters as intended and DPP only wastes the bandwidth. Also, DPP allows a message to be sent to the same vehicle more than once as we described before in subsection 5.

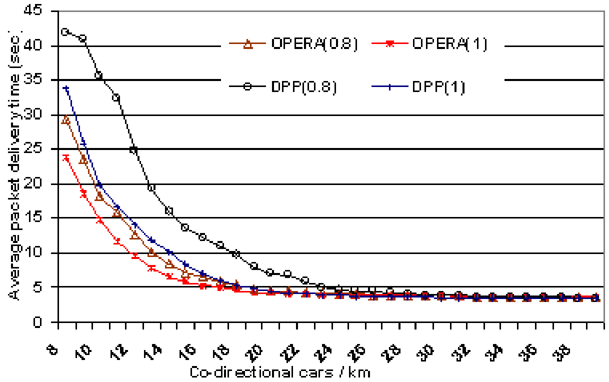


Fig. 10. *Impact of cars density on packet delivery time*

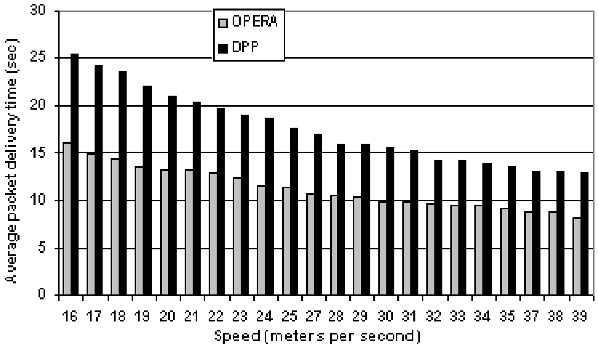


Fig. 11. *Impact of average car speed on packet delivery time*

OPERA avoids this overhead by sending a message only if it knows that it will achieve some progress along the propagation path; otherwise, it uses muling. Hence, OPERA does not send unnecessary messages. Indeed, figure9(c) shows that for a single packet, DPP would waste much more bandwidth than the clustering process.

2) Packet delivery time: Figure 10 shows the impact of traffic density on the packet delivery time in both DPP and OPERA when the ratio between oncoming traffic density and co-directional density is 1 and 0.8. The Figure shows that OPERA outperforms DPP in terms of packet delivery time. The reason is that DPP favors the oncoming direction over the co-directional direction for packet propagation while OPERA does not favor any direction, seeking instead the optimal path to propagate the packet. Also, a header car in DPP would send the packet to oncoming cluster and wait for confirmation from co-directional cluster on the road. If for any reason, this confirmation is lost or did not even exist because there is no such cluster, the header car would wait for some time before trying again.

As it turns out, as the density of the oncoming traffic decreases, OPERA will be much faster than DPP. As an illustration, for 15 cars/km in the co-directional direction, packet delivery time in OPERA is about 50% of the packet delivery time needed by DPP. Under dense traffic, both protocols would have almost the same average packet delivery time as the network would be fully connected.

Figure 11 shows the impact of average car speed on average packet delivery time when even traffic density equals 12 cars/km. Of course, for both protocols, packet delivery time would decrease as the average speed increases specially in sparse traffic where muling is the norm. For the same reasons described before, OPERA outperforms DPP under different car's speeds.

VI. CONCLUDING REMARKS

This paper has confirmed analytically the empirical findings of [7], [15] that in highway scenarios co-directional traffic is disconnected. Motivated by this, we have developed an Opportunistic Packet Relaying protocol (OPERA) where packet relaying among co-directional clusters is achieved opportunistically by a combination of data muling and local routing with the help of oncoming clusters. OPERA turns out to be optimal in the sense that no protocol can deliver a packet faster. The performance of OPERA, in terms of packet relaying time and overhead, was evaluated by extensive simulation.

REFERENCES

[1] V. Naumov and T. R. Gross, "Connectivity-aware routing (car) in vehicular ad-hoc networks," *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*, pp. 1919–1927, May 2007.

[2] A. Agarwal, D. Starobinski, and T. D. C. Little, Exploiting downstream mobility to achieve fast upstream message propagation in vehicular ad hoc networks, *Proc. IEEE Mobile Networking for Mobile Environments*, (INFOCOM/MOVE), May 2007.

[3] Car2Car Communication Consortium, http://www.car-2-car.org/ 2007.

[4] DSRC FAQ http://grouper.ieee.org/groups/scc32/dsrc/faq/ 2007.

[5] A. Festag, R. Baldessari and H. Wang, On power-aware greedy forwarding in highway scenarios, *Proc. 5-th International Workshop on Intelligent Transportation* (WIT'07), March 2007.

[6] D. E. Graham, R. L. Knuth and O. Patashnik, *Concrete Mathematics*, Addison-Wesley, Reading, Massachusetts, 1989.

[7] T. D. C. Little and A. Agarwal, An information propagation scheme for VANETs, *Proc. IEEE Intelligent Transportation Systems*, 2005.

[8] T. D. C. Little and A. Agarwal, A new information propagation scheme for vehicular networks, *Proc. 3-rd Intl. Conf. on Mobile Systems, Appplications and Service*, (MobySys 2005), Seattle 2005.

[9] M. Mabiala, A. Busson and V. Veque, Inside VANET: Hybrid network dimensioning and routing protocol comparison, *Proc. IEEE Vehicular Technology Conference*, (VTC2007-Spring), 227–232, 2007.

[10] J. Ott and D. Kutscher, Drive-thru Internet: IEEE 802.11b for automobile users, *Proc. IEEE INFOCOM*, 2004.

[11] R. C. Shah, S. Roy, S. Jain W. Brunette, Data Mules: modeling a three-tier architecture for sparse sensor networks, *Proc. 1-st IEEE Workshop on Sensor Network Protocols and Applications*, 2003.

[12] R. Stoleru, J. A. Stankovic, Probability Grid: a location estimation scheme for wireless sensor networks, *Proc. IEEE International Conference on Sensor and Ad Hoc Communications and Networks*, (SECON'04), Santa Clara, CA, 2004.

[13] M. Treibner, A. Kesting and D. Helbing, Understanding widely scattered traffic flows, the capacity drop and platoon as effects of variance driven time gaps, *Physical Review E*, 74, 2006, 1–10.

[14] US Department of Transportation, *National Transportation Statistics 2006*, US Government printing Office, Washington, DC, February 2006.

[15] N. Wisitpongphan, F. Bai, P. Mudalige, O. K. Tonguz, On the routing problem in disconnected vehicular ad hoc networks, *Proc. IEEE INFOCOM 2007*, May 2007.

[16] M. Zhaomin, Z. Hao, K. Makki, and N. Pissinou, MURU: A multi-hop routing protocol for urban vehicular ad hoc networks, *Proc. IEEE Annual Conference on Mobile and Ubiquitous Systems, Networks and Services* (MOBIQUITOUS), 2006.

[17] K. Sampigethaya, L. Huang, M. Li, R. Poovendran, K. Matsuura and K. Sezaki, CARAVAN: providing location privacy in VANET, *Proc.* 3*-rd IEEE Workshop on Embedded Security in Cars*, (ESCAR 2005), Cologne, Germany, 2005.

[18] M. Pica Ciamarra, Optimizing on-ramp entries to exploit the capacity of a road, *Physical Review E*, 72, 2005, 1–8.

[19] W. Sun, H. Yamaguchi, K. Yukimasa, and S. Kusumoto, GVGrid: A QoS routing protocol for vehicular ad hoc networks, *Proc. 14th IEEE International Workshop on QoS*, June 2006.

[20] G. Korkmaz, E. Ekici and F. Ozguner, An efficient, fully ad hoc multihop broadcast protocol for inter-vehicular communication systems, *Proc. IEEE International Conference on Communications*, (ICC'06), 2006

[21] J. Moez, M. Rabah, S.-M. Senouci, and Y. Ghamri-Doudane, GyTAR: improved greedy traffic aware routing protocol for vehicular ad hoc networks in city environments, *Proc. 3-rd ACM International Workshop on Vehicular Ad Hoc Networks*, (VANET'06), September 2006.

[22] C. Lochert, M. Mauve, H. Fuessler, and H. Hartenstein, Geographic routing in city scenarios, *SIGMOBILE Mob. Comput. Commun. Rev.*, 9(1), 69–72, 2005.