



HAL
open science

DONC: Delay-based Opportunistic Network Coding Protocol

Farhan Mirani, Anthony Busson, Cédric Adjih

► **To cite this version:**

Farhan Mirani, Anthony Busson, Cédric Adjih. DONC: Delay-based Opportunistic Network Coding Protocol. MED-HOC-NET 2013 - 12th Annual Mediterranean Ad Hoc Networking Workshop, Jun 2013, Ajaccio, France. pp.1. hal-00820815

HAL Id: hal-00820815

<https://centralesupelec.hal.science/hal-00820815v1>

Submitted on 12 Jan 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

DONC: Delay-based Opportunistic Network Coding Protocol

Farhan H. Mirani
Signals and Systems Laboratory
Supelec, France
Email: mirani@lss.supelec.fr

Anthony Busson
Laboratory of Parallel Computing
University of Lyon 1 - ENS - Inria
Email: anthony.busson@inria.fr

Cedric Adjih
Hipercom2 Team
Inria, France
Email: cedric.adjih@inria.fr

Abstract—Generally, vehicular ad hoc networks (VANETs) carry information public in nature, which benefits most of the vehicular nodes involved. Therefore broadcasting data becomes a natural choice for disseminating data in VANETs. Broadcasting also offers advantages of simplicity and flexibility, by virtue of not requiring knowledge of the actual network topology. Furthermore, so-called *delay-based* vehicular broadcast mechanisms have been proven efficient in reducing redundant packet transmissions in dense networks. However, packet losses due to the imperfect wireless medium, medium access, and high mobility decrease this efficiency dramatically, re-introducing needless transmissions.

In this paper, we propose a Delay-based Opportunistic Network Coding protocol called ‘DONC’, which combines delay-based techniques with opportunistic network coding, in order to successfully cancel this detrimental effect of losses: DONC improves dissemination of broadcast data in loss-prone VANETs and reduces packet retransmissions. We simulate DONC protocol in ns2 and compare it with classical delay-based VANET broadcast mechanisms. Results prove that DONC protocol outperforms other delay-based mechanisms, specially in the scenario of lossy VANETs.

I. INTRODUCTION

Inter-vehicular communication (IVC) is emerging as a popular solution for future road communications. IVC systems have the potential to greatly influence the road security as well as to improve traffic flow by providing the drivers with critical route information such as upcoming obstacles and weather conditions. In addition to vital information related to road security, there exists a wealth of information sources and applications that will benefit the driver or passengers in mobile vehicles such as: traffic congestion reports, variable speed limits, ambulance accommodation, construction zone warnings, local or regional business locations, gas stations, web surfing, file sharing, etc.

Due to their highly agile nature, VANETs cannot work in mere client-server type configurations. One suitable choice for such type of communication is to design VANETs as wireless ad hoc networks, where mobile nodes may come in contact for a short period of time. Owing to the fast evolution of the topology, broadcast communication may be a better fitting choice for disseminating information as opposed to unicast or multicast routing, both of which require significantly more control overhead for maintenance in view of frequent topology changes.

The basic functioning of information dissemination by broadcasting is illustrated by the following scenario: on a route, a random mobile node encounters an event which needs to be notified to its neighbors. The mobile node that encountered the event will send information in the form of broadcast transmissions, which may subsequently be rebroadcasted by the receiving nodes to propagate the information to further distance from the source vehicle in multiple hops. This mechanism is called *flooding*. In a *pure flooding* mechanism, every receiving node will rebroadcast/relay the packet back to its neighbors. Obviously, such flooding is often inefficient because in many cases one node will re-broadcast information that was already received by all its neighbors, causing packet collisions and straining the already scarce radio bandwidth.

In our work, we follow the philosophy of a family of improved broadcasting schemes, denoted *delay-based* broadcasting schemes (according to the taxonomy of the survey [1]). In delay-based broadcasting, each node that receives a downstream packet does not rebroadcast it immediately. Instead, it starts a calculated timer governed by a ‘delay’ parameter. The value of the delay parameter, takes into account the current distance between the source and the receiving vehicles, such that the farther the receiver is from the source, the shorter the delay will be. Formally, the timer is expressed as some decreasing function of the distance. If by the end of the timer, the node does not receive the same packet from the opposite direction, only then it is allowed to rebroadcast. Consequently, the farthest vehicle to have received a packet will have the shortest delay value of all, and therefore must be chosen as the relaying vehicle. However, if the node receives the same packet as an upstream packet before its timer expires; it assumes that a vehicle further behind received the same packet and broadcasted before it. Thus, this upstream packet is considered an implicit acknowledgement at the receiving node and consequently the node cancels its currently running timer.

Figure 1 shows an example of a small vehicular network. Let us assume that the average broadcast coverage zone of a node is 3 vehicles from either side. From the example in the figure 1, vehicular node ‘e’ broadcasts a packet which is received by nodes ‘b’, ‘c’ and ‘d’. All three receiving nodes will then start a timer, where node ‘d’ will have a timer with the largest delay value and node ‘b’ will have a timer with

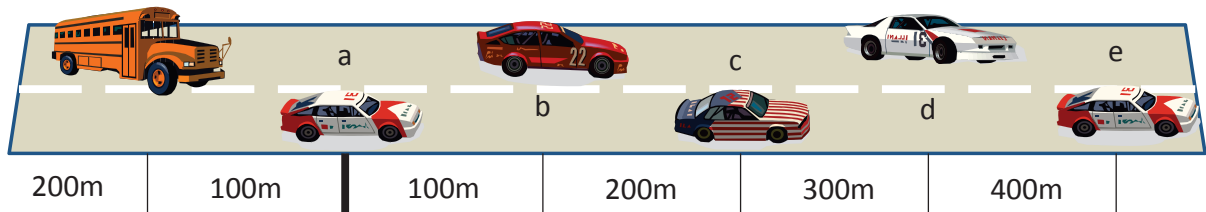


Fig. 1: An example vehicular network

the smallest delay value. Out of these three receiving nodes, the timer for node ‘b’ will expire first and therefore node ‘b’ will broadcast the packet. Given the nodes ‘c’, ‘d’ and ‘e’ are in coverage range of node ‘b’, they will receive this packet as well. Upon reception, these nodes will cancel their running timers, considering the received upstream packet as an implicit acknowledgement.

This strategy offers the optimal performance in terms of redundancy when the radio environment is perfect (a perfect and identical radio range around each transmitter), as there are exactly two receptions per node (one upstream and one downstream). Moreover, in its simplest form, this mechanism can be implemented without any control packets, and can be entirely stateless. Therefore, from the point of view of complexity and redundancy this protocol/mechanism is the ideal protocol to disseminate information in VANET. Nevertheless, when considering more realistic radio environment, where broadcast packets can be lost for all or a part of the neighbors, the delay-based mechanism does not perform so well [2]. Transmissions, also used as implicit acknowledgements, that are not properly received generate useless retransmissions causing redundancy. For certain radio models, the redundancy may be very high making this approach unsuitable.

To combat this effect, we combine such a delay-based scheme with *network coding* [3]. Network coding is a recent transmission paradigm where intermediate nodes in the *network* can *code* different incoming packets instead of just forwarding them as-is. Network coding is known to yield capacity gains and to improve resilience to losses. We show that an accurate combination of delay-based scheme and Network coding can make this approach close to its optimal performances even in realistic/lossy radio environments.

A. Contributions

This paper presents DONC, a Delay-based Opportunistic Network Coding protocol. DONC exploits the broadcast nature of the wireless medium to implement a distributed network coding scheme specifically tailored for lossy VANETs. Instead of transmitting individual packets as received, DONC linearly encodes the original packets into coded packets before a (delay-based) retransmission. Decoding a coded packet becomes possible when enough decoding information is received at a node. Upon decoding, original packets are extracted and only then these packets are allowed to be further retransmitted. DONC features six important properties and contributions:

- Delay-based broadcast: this optimized flooding with implicit acknowledgements is the basis for DONC. Because a coded packet combines several packets of sources, the entire notion of “implicit acknowledgement” has to be transposed to the context of network coding (which packets are acknowledged). DONC achieves this objective by acutely integrating the following mechanism, Stop-And-Go.
- *Stop-and-Go*: a node will retransmit coded packets only after it has been able to decode them. This yields more control on the flooding process, and also signals decoded packets.
- Low complexity: DONC is entirely designed around simplicity, and minimizes the conceptual overhead. One major contribution is the precise combination of different, individually simple, mechanisms in a consistent protocol. As a result, DONC does not require any complex coordination of nodes, knowledge of the topology (neighbors, density of the network, etc.), specific signalling, dedicated MAC mechanisms, nor any specificities beyond basic packet transmission.
- Efficiency: despite its low complexity, DONC may still reach high efficiency, as evidenced in our performance evaluation, section IV. For instance, when testing on the ideal boolean model, the protocol is within a few percents of the optimal (minimal) number of transmissions.
- Resilience to losses: with network coding, different nodes may still decode coded packets subject to different loss patterns without requiring retransmissions on a per-packet basis (which is more expensive).
- Resilience to losses in overhearing: one retransmission will act as an implicit acknowledgment for several source packets instead of just one, increasing the probability of self-elimination in lossy settings, hence performance.

B. Organization

The rest of this paper is organized as follows: Section II discusses the general background of our work, including its context and fundamental ideas. In section III, we propose an efficient data dissemination protocol for lossy VANETs called DONC protocol. Performance evaluation of the proposed protocol is presented in section IV and before concluding our paper in section VI, section V reviews some important works proposed on broadcast in vehicular networks as well as on network coding.

II. GENERAL BACKGROUND

A. Context and Assumptions

Our context here is of a completely distributed vehicular network (preferably V2V), where each vehicle is a mobile node in a vehicular ad hoc network (ref. figure 1). These vehicles are configured to transmit in broadcast mode and may transmit diverse information to other neighboring vehicles regarding road conditions, weather updates, accident warnings, traffic information, advertisements, etc. Due to high vehicle mobility, we assume the network conditions may vary from stable and predictable to highly lossy and unpredictable.

In a general VANET, a vehicle may receive either *downstream* or *upstream* packets: in the example in figure 1, for node ‘c’, all the packets received from nodes ‘a’ and ‘b’ are upstream and all the packets from nodes ‘d’ and ‘e’ are downstream. In order to keep the context of our work simple at this point, we consider that the propagation of data is carried out in only one direction, in the upstream direction (from right to left in figure 1, and with downstream packets): this direction is the most relevant for typical VANET applications, since it can convey information on the environment that will be met later by the vehicles (places of interest, incidents, ...).

Finally, we assume all the vehicular nodes involved are equipped with GPS receivers so that each receiving vehicle can deduce its distance from the transmitting vehicle.

B. Network Coding Operation

We assume every vehicle in the network can not only send and receive packets of data but can also perform network coding operations. Network coding allows a node to “mix” two or more data messages into a coded packet and broadcast it. This mixing of data messages is referred as ‘packet encoding’. Upon reception of a data packet, the receiving vehicle tries to ‘decode’ it with the information it currently possesses. For the sake of simplicity, we assume a data packet to be a container of one or more unique data messages. If a data packet contains only one message, it is considered not-coded and hence its contents can be retrieved on its own, without any further information. However if a packet contains more than one messages in it, the data packet is considered to be encoded and it can only be decoded if sufficient decoding information is present at the receiving node.

For detailed information about network coding operations in our work, readers are referred to related work (section V) and cited works: [4] for a general description of the theory and benefits of network coding, and MORE [5] and OMNC [6] which explain in detail network coding models identical to ours. In summary: packets are supposed to be of identical size (concatenated and padded as necessary), coded packets are generated from source packets with linear combinations through random linear coding [7], decoding is performed with Gaussian elimination and encoding vectors are inserted in headers [8] (with a mapping from packet indices to node and packet identifiers; finding efficient mappings is not addressed by this article, see [9] for instance).

III. DONC PROTOCOL

DONC, or Delay-based Opportunistic Network Coding protocol fuses the idea of the popular delay-based broadcasting mechanism for VANETs with the principles of network coding, which allow to improve network throughput and reduce packet loss in error and collision prone wireless networks.

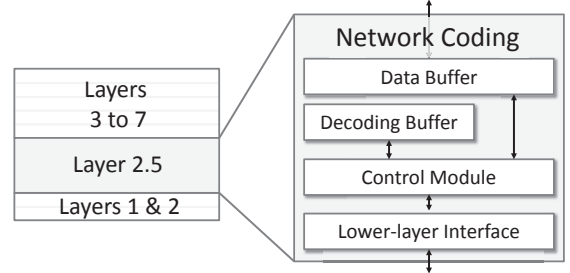


Fig. 2: Network Coding (NC) Layer - Decomposition

A. Protocol Semantics and Architecture

1) **Overview:** In order to implement network coding with a delay-based broadcasting mechanism, we propose to introduce an intermediate layer in the protocol stack called the Network Coding (NC) layer, as shown in the figure 2. In the NC layer, we introduce two distinct buffers to store incoming data packets: the Data-Buffer and the Decoding-Buffer.

- **Data-Buffer:** It is a buffer that stores decoded/not-coded data packets.
- **Decoding-Buffer:** It is a temporary buffer to store incoming coded data packets which cannot immediately be decoded at the receiving vehicle with Gaussian elimination and require additional coded data packets.

The NC layer also has a Control Module and a Lower Layer Interface module. The Control Module is responsible for all the operations of the protocol, which include fetching data from and to the Data-buffer and Decoding-buffer, running packet decoding procedures, keeping track of timers, etc. Lower Layer Interface module provides interface between NC layer and layer-2.

The functioning of DONC is based on the following principles:

- Coded packets stay in the decoding buffer until decoded,
- Reception from upstream of a given packet acts as an implicit acknowledgement and any scheduled (re)transmission of the packet is canceled,
- Decoded/non-coded packets are transmitted after an initial delay-based timer (unless canceled),
- Decoded/non-coded packets are later retransmitted until implicit acknowledgement,
- Any transmission of decoded/non-coded packets is performed by first computing a linear combination of such available packets (network coding).

2) **Data Structures:** Every node in the network has a reference table (Ref-Table), which keeps information regarding packet transmissions, timers and acknowledgements. It has three fields: Msg-Ids, Ack-Flgs and Exp-Timers corresponding

to each data message in the Data-buffer (ref. table I). These entries are updated every time a vehicle has some activity (transmission or reception). Since Ref-Table is a reflection of the current contents of Data-Buffer, the size of Ref-Table is equivalent to that of the Data-Buffer.

- **Msg-Id:** It is a unique ID given to each message before it is encoded in a data packet. A data packet may contain many messages and hence many Msg-Ids. Each vehicle uses a unique tag (originator IP address and packet sequence number) to differentiate from the packets originating at neighboring nodes.
- **Ack-Flg:** It is a flag which indicates if a particular message has been acknowledged or not.
- **Exp-Timer:** Every message in the data buffer is associated an expiry time, after which the message should be (re)broadcasted. Exp-Timer field indicates the time when the vehicle wants this message to be (re)broadcasted.

Msg-Id	Ack-Flg	Exp-Timer
01	1	1.021554 sec
02	0	1.121245 sec
03	0	1.245644 sec

TABLE I: Example of a Ref-Table

Implicit Acknowledgments: Ack-Flg Update

Our protocol operates with implicit acknowledgements exclusively. When a vehicle receives a second copy of a particular packet from one of its neighboring vehicles, it checks if it is a downstream or upstream packet. In case it is a downstream packet, it is considered to be an unnecessary retransmission and hence is discarded upon reception. However if it is an upstream packet, it means the transmitting vehicle has just forwarded this packet further ahead, so the node considers this an implicit acknowledgement. The value '0' in the Ack-Flg field indicates that an acknowledgement is still awaited for the corresponding message. '1' means the message is already acknowledged and accounted for.

Timer: Exp-Timer Update

The initial value to be given to Exp-Timer is calculated as:

$$\text{Exp_Timer} = t + \text{Delay} \quad (1)$$

where ' t ' is the current system time and Delay is a decreasing function of the distance between the transmitter and the receiver. Delay is '0' if the message is generated by the broadcasting vehicle itself, which means this message needs to be transmitted immediately. After the first expiry of the timer, the timer becomes a timeout ensuring retransmission until an implicit acknowledgement is received (or a limit is reached).

3) **Semantics:** DONC semantics rely on three kinds of events: local application packet handling, packet reception and coded packet transmission.

Local Application Packet Handling

- Every time there is some information to be broadcasted from a local application, the NC layer receives it from higher layers in the form of an outgoing packet. The network Coding layer first sends it to be stored in the Data-buffer. The NC layer then creates an entry in the reference table corresponding to the newly received message with its message ID number, flag set to '0' and expiry timer set to current system time. Finally, the NC layer starts the packet transmission routine. Flow-diagram of this process can be seen in figure 3a.

Packet Reception Similarly, a flow-diagram for incoming packets is shown in figure 3b. Every time a vehicle receives a packet from a neighboring vehicle, the NC layer receives it from the lower layers. The NC layer first checks if the packet can be decoded with the information currently available at the vehicle (Data-buffer and Decoding-buffer).

- If it cannot be decoded, the NC layer sends the coded packet to be stored temporarily in the Decoding-buffer and wait for more information to be received.
- On the other hand if the received packet can be decoded, the NC layer extracts individual data messages from it. Upon extraction, it is first checked whether each message is innovative or not.
 - The innovative messages are sent to be stored in the Data-buffer and their respective reference table entries are created (Msg-Id, Ack-Flg and Exp-Timer). Every time an innovative message is added to the Data-buffer, the NC layer checks if there are any coded packets in the Decoding-buffer that can now be decoded with the help of this newly received innovative message. If a coded packet is decoded, this operation is repeated.
 - The non-innovative messages may either be acknowledgements or unnecessary retransmissions. To verify which category of the two they fall in, the NC layer checks if the incoming packet was a downstream or upstream packet. A downstream packet will mean the message was a retransmission, and therefore, is immediately discarded without further inquiry. If however, the message was received in an upstream packet, it means this message is an acknowledgement of a previous broadcast. Therefore, NC layer will update the reference table entry corresponding to the received acknowledgement by flipping its flag from '0' to '1'.
- After every activity on the Data-buffer, the buffer is sorted in ascending order by the values of its Exp-Timer field, so that the elements with the smallest values are on the top. The NC layer then assigns the timer the value of the first element from the top of the reference table whose Ack-Flg is not '1', i.e., the corresponding message is not already acknowledged. From the example given in table I, although the Exp-Timer value of message '01' is smaller than that of message '02', but the vehicle timer skips the first value because the message is already acknowledged

and takes the value of Exp-Timer of message ‘02’ instead.

Coded Packet Transmission

On a vehicle, NC layer starts the packet transmission routine (ref. flow-diagram in figure 3c) as soon as its timer expires. First, the NC layer picks the first ‘N’ messages from the reference table whose Ack-Flgs are ‘0’. It then encodes these ‘N’ messages in a coded data packet, before sending it to the lower layers for broadcast. NC layer then updates the Exp-Timer values in the reference table for the corresponding messages as:

$$\text{Exp_Timer} = t + \text{Ret_Timeout} \quad (2)$$

where ‘Ret_Timeout’ is the retransmission timeout, a constant set to a value of the order of magnitude of the time necessary for a packet to be received and acknowledged by at least one of the neighboring vehicular nodes.

IV. PERFORMANCE EVALUATION

In this section, we evaluate the performance of DONC protocol by simulation. We use Network Simulator 2 (ns2) [10], which is an open source discrete event network simulator. We compare the performance of DONC protocol with a standard delay-based broadcast mechanism. To avoid unnecessary repetition of words, we call it the SDB (Standard Delay-based Broadcast) protocol. The presented results illustrate how combining network coding with a simple delay-based VANET broadcasting mechanism may help improve its performance, specially in adverse network conditions.

A. Simulation Scenario

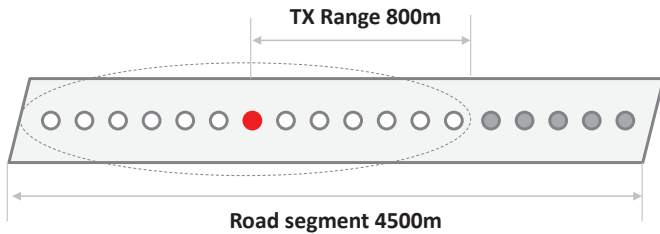


Fig. 4: Simulation Topology

The topology we chose to test DONC protocol is as shown in figure 4. It consists of a fixed road segment of 4.5 kilometers approximately. All the vehicular nodes are equipped with radio equipment on the specifications of IEEE 802.11p standard. IEEE 802.11p is an enhancement to the IEEE 802.11 standard destined at adding Wireless Access in Vehicular Environments (WAVE) [11]. The vehicles in our topology are configured to transmit in a radius of 700 meters approximately, which corresponds to the 802.11p standard. Furthermore, in order to obtain results that are easier to interpret, we assume that the vehicles are regularly distributed (e.g identical inter-vehicle distance) and for each test, we vary the node density (from 5 veh/km to 45veh/km). We simulate a broadcast of 10 packets ($pkt_1, pkt_2, pkt_3, \dots, pkt_{10}$) from 10 first nodes (n_0, n_1, \dots, n_{10}), chosen randomly with a time interval of 1ms between each

broadcast. Other traffic patterns have been considered, and lead to the same performances. They are not shown here for the sake of clarity.

In order to test DONC protocol in different wireless scenarios, we configured three distinct FER (Frame Error Rate) models (shown in figure 5), one for each rural (scarce population), complex multi-path fading environments and ideal (no-loss).

- The 2RM FER model was proposed in [12]. It is a measurement based model of the frame error process in rural setting. The model takes into account 802.11p wavelength, heights, distances, antenna gains, frame length, etc.
- The Rayleigh FER model is destined for more complex radio environment in presence of multi-path fading [13]. In our case, the Rayleigh model serves as the ‘worst case scenario’ where FER changes frequently and does not present a definite threshold function. From the figure 5, it can be noted that packet reception rate for Rayleigh decreases quickly, even for small distances.
- The Boolean FER model is a custom-built ideal radio model designed to compare with the performances of more real-like radio models with little and heavy radio losses. The packet reception rate in this model is 1 for x in $[0, 700]$ and 0 for $x > 700$.

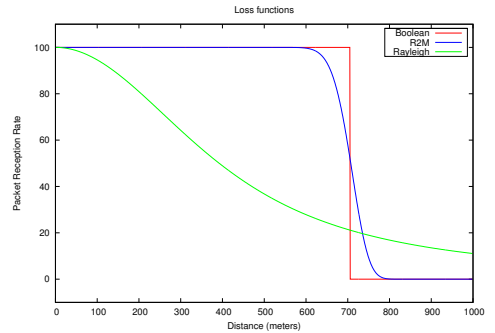


Fig. 5: Boolean, 2RM and Rayleigh loss functions

For the delay selection in (1), the delay is a linear function of the distance with: $\text{Delay} = \beta - \alpha \times \text{distance}$ (for our simulations, $\beta = 0.36$ and $\alpha = 0.0005$ with distances expressed in meters and times in seconds).

B. Simulation Results

In the simulations, almost all nodes (i.e. excluding a few downstream sources that do not receive other sources upstream) ultimately receive and decode the source packets. Then, the meaningful performance metric is the amount of redundancy, characterized by the ratio of average number of received packets per source packet (lower is better). In an ideal dense linear network without losses, the average number of receptions would be equal to 2: one reception from downstream and one reception from upstream (as it is further propagated by one repeater).

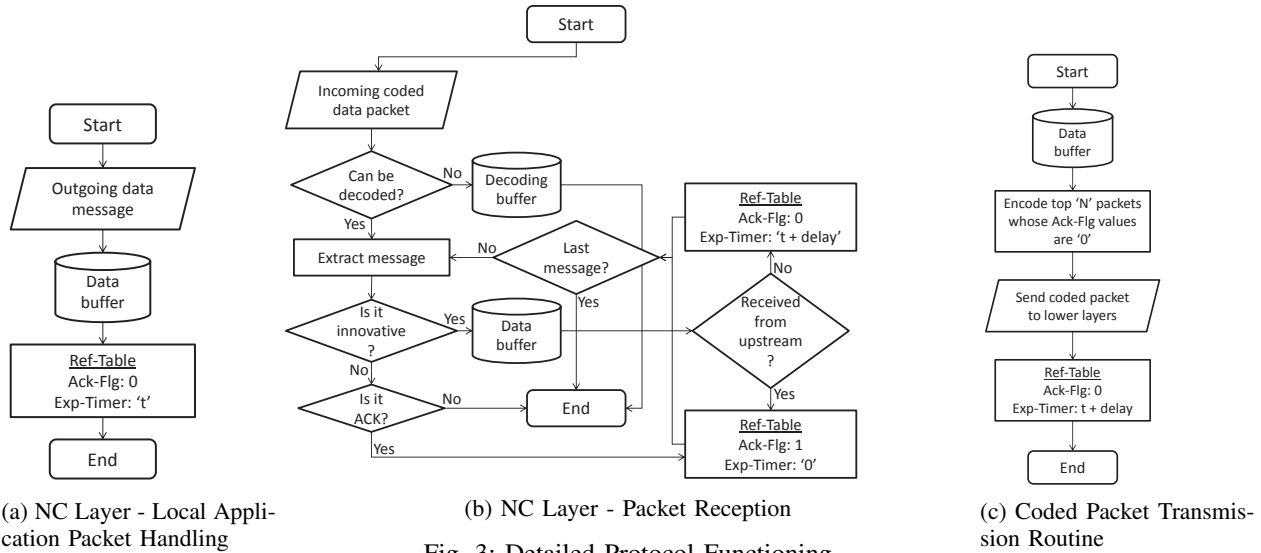


Fig. 3: Detailed Protocol Functioning

Figures 6a and 7a compare the performance of the DONC protocol with the SDB protocol in 2RM loss model. It clearly appears that for the vehicle densities ranging from 5 to 45 veh/km, the average number of packet receptions and transmissions per vehicle with the DONC protocol remain lower than with SDB. This is because the SDB protocol suffers increased packet redundancy to counter the effects of packet loss in VANETs, thereby increasing the network traffic as well as total channel occupancy of the network. On the other hand, the DONC protocol uses principles of network coding to reduce the number of redundant packet transmissions by encoding multiple messages in a coded packet. Lower unneeded packet redundancy with the DONC protocol translates into lower channel occupancy and lower wireless data traffic for the same amount of information to be communicated.

Figures 6b and 7b indicate that DONC performs even better in highly congested and lossy environments. The Rayleigh loss model is adapted to complex urban centres where wireless medium quality is poor and wireless signals may be susceptible to heavy multipath fading. It can be seen in these figures (6b and 7b) that the difference in performance of DONC over SDB is even greater than it was for 2RM model. This is because the Rayleigh loss model simulates much higher packet loss rates, and thus SDB increases its packet redundancy to cover for the increased packet loss. On the other hand, DONC encodes multiple messages in individual packets to reduce the packet redundancy and achieve better network performance.

While these results are sufficient for real environments where losses occur mainly due to average/poor radio coverage, we should also make sure that DONC protocol performs equally, if not better than the SDB protocol in an ideal environment with perfect radio reception, in order to check for any overhead. Figures 6c and 7c present a comparison between DONC and SDB in an ideal environment (Boolean loss model), where the only losses possible are due to the collisions among different broadcasts. It can be seen in the figure that the performance of the DONC protocol in an ideal

environment is as good as for the SDB protocol, both staying close to the ideal value of 2.

V. RELATED WORK

This section is further divided into two subsections. First, we outline some important works on data dissemination and broadcast in VANETs and the second subsection briefly overviews network coding and some related works.

A. VANET Relaying

Packets travel through multi-hop broadcast networks by way of flooding. Ideally, a vehicular node (source) will transmit an information packet in broadcast mode and all the neighboring vehicles (receivers) in its transmission vicinity will receive the packet. Each of these receiving vehicles will act as relaying nodes and rebroadcast the received packet to their neighbors and so forth. In this way, the information packet may propagate through the vehicular network.

However, a *vanilla* flooding mechanism is most likely to be inefficient as every single vehicle that receives a packet will rebroadcast it, causing redundant transmissions wasting scarce radio resources. There are several main techniques used to solve the above mentioned problems including: optimized flooding (with connected dominated sets), probabilistic broadcasting and delay-based broadcasting.

In the connected dominating set approach, a (precomputed) subset of the nodes retransmits the messages. In probabilistic broadcasting, different rebroadcast probabilities (also called forwarding probability) are assigned to each vehicle in the receiving range of a source vehicle, and again only a fraction of all the receiving vehicles rebroadcast a received packet, but this time, the set of such nodes is probabilistically determined. An example is provided by the authors of [14] who propose a distance-based probabilistic broadcasting mechanism. Upon reception of a packet, the mechanism determines its forwarding probability based on the distance between the receiving and source vehicles. Farther the receiving vehicle, higher the probability of rebroadcasting.

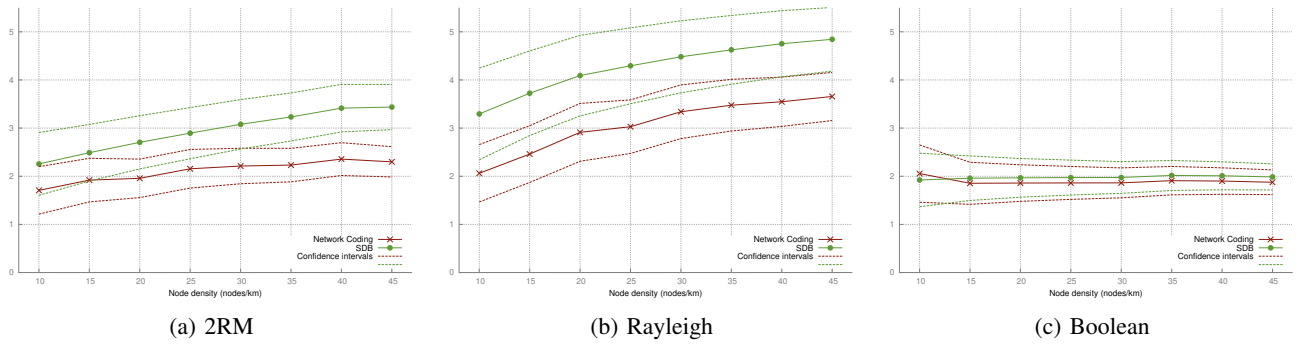


Fig. 6: Average receptions per vehicle per source packet

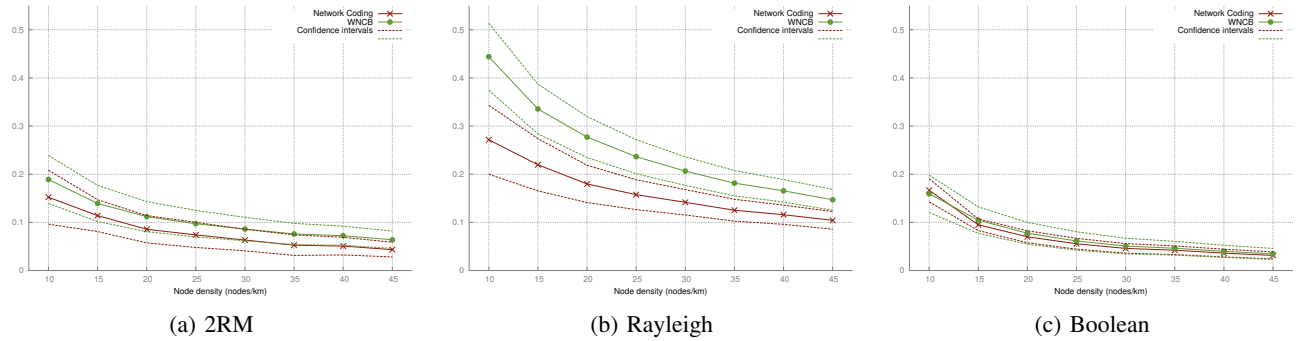


Fig. 7: Average transmissions per vehicle

Just as probabilistic broadcasting is based on calculating an optimal value for forwarding probability, delay-based broadcasting mechanisms are based on determining the optimal delay on each receiving node before the packet can be rebroadcast. A prime example is Multi-hop Vehicular Broadcast (MHVB) [15].

Urban Multi-hop Broadcast (UMB) [16] proposes to prioritize the vehicles by the position of their road segments in reference to the transmitting vehicle. When a packet is to be broadcasted, UMB transmitter first sends a request-to-broadcast (RTB) control message. Upon reception of a RTB, each receiving vehicle will start transmitting a *blackburst* signal, for a time that will be determined by its distance from the transmitting vehicle. The farthest vehicle to have received the RTB will transmit the longest *blackburst*, and therefore is chosen implicitly as the forwarding node. Authors in [17] present Efficient Directional Broadcast (EDB). Although similar in approach to UMB, EDB does not use RTB/CTB control messaging. EDB makes use of the directional antennas to distinguish the two directions of packet propagation (forward and backward) and has a unique waiting time calculating function.

B. Network Coding

Originally proposed by R. Ahlswede in [3], the idea of network coding breaks the long established assumption that an intermediate network node has to forward the exact same data it received and spawned a large number of results and protocols.

Fragouli et al., in their work [18] presented distributed network coding algorithms achieve energy savings (minimize

the number of retransmissions). Katti et al. proposed COPE [19]. Although it is a network coding based unicast routing protocol, COPE uses key properties of wireless communications, which also encompass communication in broadcast mode. The three basic techniques in COPE are based on: (a) snooping or overhearing all the packets being transmitted in the wireless broadcast medium that the node can overhear, (b) encoding packets in a combination with the guarantee that the combination will be decoded at the destination and (c) keep track of all the packets the neighbors have received so far. This is achieved with the reception reports, broadcasted by each node periodically.

A family of protocols have exploited network coding for its ability to let network information flow along multiple paths in parallel. Chachulsky et al. proposed MORE [5], which is a distributed multipath opportunistic routing (OR) mechanism. MORE proposes that multiple intermediate nodes between a source and a destination combine ‘native packets’ of data into linearly independent coded combinations (packets). These coded versions are then forwarded towards their common destination. Since these coded packets are linearly independent of each other, it suffices to deliver any ‘ N ’ combinations to the destination to successfully decode ‘ N ’ native packets of data. Results prove that MORE improves communication performance. Zhang et al. [6] present Optimized Multipath Network Coding (OMNC), which optimizes MORE to work in a distributed environment by assigning each node an encoding and broadcast rate in a decentralized manner. OMNC explores the broadcast nature of the wireless medium and its possible path diversity, and takes advantage of network coding to adapt to lossy environment by controlling the rate at which native

packets may be encoded and/or broadcasted. DRAGONCAST [20] proposes an adaptive broadcast protocol.

For VANETs themselves, the idea of COPE was adapted in [21], which proposed two distinct queues for coding/decoding packets, each in a different direction of the traffic on a road and looking for opportunities to broadcast an XOR-ed combination which may contain packets useful to both directions simultaneously. Similarly, authors in [22] propose CODEB, a network coding based broadcast protocol. Also inspired by COPE, CODEB covers broadcasting scenarios for ad hoc wireless networks, in which opportunistic coding is performed not just for intended next-hop neighbor but for all the neighbors surrounding a node. Another relevant work in the area is presented in [23] which reduces the number of transmissions required to flood packets in an ad hoc wireless network. The principle difference between DifCode and CODEB is that CODEB required all of its receivers to instantaneously decode the incoming combinations.

An additional network coding mechanism is the use of “Symbol-Level Network Coding” (SLNC) [24] to improve performance in lossy/noisy wireless networks when receiving a packet with bits errors, by essentially using the “parts” without errors. However in the 802.11p context, benefits are partly mitigated by the convolutional coding (FEC) of 802.11p, and SLNC may require some radio chipset support. An example of VANET protocol using SLNC is CodeOn [25].

For further details on broadcast in VANETs, network coding and its working principles, readers may refer directly to [3],[7],[8],[2] and [1].

VI. CONCLUSION

Delay-based VANET broadcast protocols provide an efficient way to reduce the number of unnecessary transmissions by using repeated packets as implicit acknowledgements. However, these protocols do not perform well in lossy VANETs as the acknowledgements may be lost and not be delivered to a vehicle, forcing unnecessary packet repetitions. In this paper, we proposed a solution to this issue: DONC, a simple yet efficient protocol that uses network coding opportunistically to counter the adverse effects of such packet losses. We described the details of the protocol. We simulated DONC and compared it with a classical delay-based VANET broadcast protocol. The results of these simulations illustrate that the fact that DONC outperforms delay-based mechanisms in lossy VANETs; in addition its performance remains steady in less lossy scenarios. Ongoing and future work include a model to estimate the performance of DONC in a more general context. This model will aim to provide insights on the protocol parametrization, in particular on the trade-off between the timer function and the ratio of coded/uncoded packets.

REFERENCES

- [1] S. Panichpapiboon and W. Pattara-atikom, “A Review of Information Dissemination Protocols for Vehicular Ad Hoc Networks,” *Communications Surveys Tutorials, IEEE*, vol. 14, no. 3, pp. 784–798, 2012.
- [2] A. Busson, “Analysis and simulation of a message dissemination algorithm for vanet,” *International Journal of Communication Systems*, vol. 24, no. 9, pp. 1212–1223, 2011.
- [3] R. Ahlswede, N. Cai, S.-Y. Li, and R. Yeung, “Network information flow,” *Information Theory, IEEE Transactions on*, vol. 46, no. 4, pp. 1204–1216, jul 2000.
- [4] e. Al Agha, K., *Network Coding*. Wiley, 2012.
- [5] S. Chachulski, M. Jennings, S. Katti, and D. Katabi, “MORE: A Network Coding Approach to Opportunistic Routing,” Massachusetts Institute of Technology, Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Tech. Rep., June 2006.
- [6] X. Zhang and B. Li, “Optimized Multipath Network Coding in Lossy Wireless Networks,” in *Proceedings of the 2008 The 28th International Conference on Distributed Computing Systems*, ser. ICDCS '08. Washington, DC, USA: IEEE Computer Society, 2008, pp. 243–250. [Online]. Available: <http://dx.doi.org/10.1109/ICDCS.2008.45>
- [7] T. Ho, M. Mard, R. Koetter, D. R. Karger, M. Effros, J. Shi, and B. Leong, “A random linear network coding approach to multicast,” *IEEE TRANS. INFORM. THEORY*, vol. 52, no. 10, pp. 4413–4430, 2006.
- [8] P. A. Chou, Y. Wu, and K. Jain, “Practical network coding,” 2003.
- [9] L. Keller, M. Jafari Siavoshani, C. Fragouli, K. Argyraki, and S. Diggavi, “Identity Aware Sensor Networks,” in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, 2009.
- [10] “The Network Simulator 2 (NS-2).” [Online]. Available: <http://www.isi.edu/nsnam/ns>
- [11] “Ieee 802.11p: Wireless access in vehicular environments (wave).” [Online]. Available: <http://www.ieee802.org>
- [12] P. Barsocchi, G. Oliveri, and F. Potorti, “Frame error model in rural wi-fi networks,” in *Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks and Workshops, 2007. WiOpt 2007. 5th International Symposium on*, april 2007, pp. 1–6.
- [13] K.-L. D. Du and M. N. S. Swamy, *Wireless Communication Systems*. Cambridge University Press, 2010.
- [14] N. Wisitpongphan, O. K. Tonguz, J. S. Parikh, P. Mudalige, F. Bai, and V. Sadekar, “Broadcast storm mitigation techniques in vehicular ad hoc networks,” *Wireless Commun.*, vol. 14, no. 6, pp. 84–94, Dec. 2007. [Online]. Available: <http://dx.doi.org/10.1109/MWC.2007.4407231>
- [15] T. Osafune, L. Lin, and M. Lenardi, “Multi-hop vehicular broadcast (mhvb),” in *ITS Telecommunications Proceedings, 2006 6th International Conference on*, june 2006, pp. 757–760.
- [16] G. Korkmaz, E. Ekici, F. Özgüner, and U. Özgüner, “Urban multi-hop broadcast protocol for inter-vehicle communication systems,” in *Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks*, ser. VANET '04. New York, NY, USA: ACM, 2004, pp. 76–85. [Online]. Available: <http://dx.doi.org/10.1145/1023875.1023887>
- [17] D. Li, H. Huang, X. Li, M. Li, and F. Tang, “A distance-based directional broadcast protocol for urban vehicular ad hoc network,” in *Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007. International Conference on*, sept. 2007, pp. 1520–1523.
- [18] C. Fragouli, J. Widmer, and J.-Y. Le Boudec, “A Network Coding Approach to Energy Efficient Broadcasting: From Theory to Practice,” in *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, april 2006, pp. 1–11.
- [19] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, “Xors in the air: Practical wireless network coding,” *Networking, IEEE/ACM Transactions on*, vol. 16, no. 3, pp. 497–510, june 2008.
- [20] S. Cho and C. Adjih, “Wireless broadcast with network coding: Dynamic rate selection,” in *the 7th IFIP Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, Jun. 2008.
- [21] L. Haojie, Z. Dongdong, and Y. Chen, “Local-directed network coding in vehicular ad-hoc networks,” in *Intelligent Signal Processing and Communication Systems (ISPACS), 2010 International Symposium on*, dec. 2010, pp. 1–4.
- [22] L. Li, R. Ramjee, M. Buddhikot, and S. Miller, “Network Coding-Based Broadcast in Mobile Ad-hoc Networks,” in *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*, may 2007, pp. 1739–1747.
- [23] N. Kadi and K. A. Agha, “Mpr-based flooding with distributed fountain network coding,” in *Ad Hoc Networking Workshop (Med-Hoc-Net), 2010 The 9th IFIP Annual Mediterranean*, june 2010, pp. 1–5.
- [24] S. Katti, D. Katabi, H. Balakrishnan, and M. Medard, “Symbol-level network coding for wireless mesh networks,” *SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 4, pp. 401–412, Aug. 2008. [Online]. Available: <http://doi.acm.org/10.1145/1402946.1403004>
- [25] M. Li, Z. Yang, and W. Lou, “Codeon: Cooperative popular content distribution for vehicular networks using symbol level network coding,” *IEEE Journal on Selected Areas in Communications*, pp. 223–235, 2011.