3rd International Workshop on Smart Vehicles: Connectivity Technologies and ITS Applications (SmartVehicles 2016) co-located with WoWMoM 2016, Coimbra, Portugal, June 2016

Impact of Decentralized Congestion Control on Contention-based Forwarding in VANETs

Sebastian Kühlmorgen, Andreas Festag, Gerhard Fettweis
Vodafone Chair Mobile Communications Systems, Technische Universität Dresden, Germany
{sebastian.kuehlmorgen,andreas.festag,gerhard.fettweis}@tu-dresden.de

***Abstract*— Vehicular ad hoc networks based on IEEE 802.11 OCB are expected to suffer from channel congestion, which will be critical for safety applications when the channel becomes saturated. Therefore decentralized congestion control (DCC) is needed that keeps the channel load under a pre-defined threshold. In the European system for vehicular communication, a DCC-gatekeeper has been inserted into the protocol stack that throttles the data traffic generated or forwarded by the vehicle. This DCC-gatekeeper affects the functionality of the contention-based forwarding (CBF) and degrades its performance. In this paper we evaluate the performance of CBF for several protocol variants – DCF, EDCA and DCC-gatekeeper – under congestion-free and saturated channel conditions. The results show that the standard DCC-gatekeeper hampers the functionality of CBF even for small node densities. We enhance the DCC-gatekeeper-based approach to overcome this issue and evaluate the protocol variants in terms of reliability, latency and channel load in a freeway scenario with realistic trace-based mobility.**

I. Introduction

Inter-vehicle communication extends the range of perception of the drivers and vehicle on-board sensors, such as radar or camera. The data exchange among vehicles and with roadside stations is expected to significantly enhance road safety, traffic efficiency and driving comfort. A proposed system for vehicular ad hoc networks (VANETs) is based on Wi-Fi and operates in the 5.9 GHz frequency range dedicated for road safety and traffic efficiency. ETSI, the European Telecommunications Standards Institute, has defined a release 1 of standards that covers the access technology ITS-G5[1], ad hoc networking using geographical information with multi-hop forwarding algorithms, and message types at the facilities layer, such as the Cooperative Awareness Message (CAM) and the Decentralized Environmental Notification Message (DENM) [1].

It is well known that in IEEE 802.11 OCB-based VANETs, the potentially large number of vehicles, the high data load generated by applications and multi-hop forwarding can result in data congestion in the bandwidth-limited wireless channels. The inherent characteristics of IEEE 802.11 OCB imply that the load on the wireless channel need to be kept low – practically below a pre-defined threshold – in order to mitigate packet collisions in the wireless channel and a potentially unlimited medium access delay.

For this reason decentralized congestion control (DCC) is introduced and various proposals for DCC have been made. For instance, DCC algorithms can control the transmission interval of messages at the facility layer [2], [3] in order to reduce the number of generated messages at high channel load. Other algorithms control the transmit power to adapt the transmission range and therefore the number of reachable nodes [3], [4]. Further options include the increase of the data rate if the channel has an appropriate quality or the transmitter can vary the carrier sense threshold [5]. A survey of DCC methods can be found in [6]. The DCC framework standardized by ETSI has been analyzed in [7].

Recently, a DCC-gatekeeper has been proposed for the DCC solution standardized by ETSI TS 103 175 V1.1.1.[2] In the protocol stack it is inserted between the NET and MAC layer. The DCC-gatekeeper reduces the generated data traffic and enforces that a vehicle adheres to data traffic limits (see ETSI TR 101 612 V1.1.1 for an architecture description). Furthermore, an additional delay depending on the channel load is included. The main advantage of the DCC-gatekeeper compared to other solutions is the fact that all data traffic in a node is controlled by a single entity, including the data packets that are forwarded. DCC algorithms that control the message rate at the facility layer cannot cope with forwarded packets.

In general, for DCC-gatekeeper flow control two approaches exist: reactive and adaptive. In the reactive approach a state machine compares the channel load with a predefined look-up table and sets the values, such as packet rate or transmit power, accordingly. Early studies indicate that with this scheme the system tends to oscillate among states [8]. It has been shown that the adaptive approach, such as LIMERIC [2], outperforms the reactive approach [8] and exhibits a stable state behavior. Therefore, in this study we use LIMERIC for the packet rate computation of the DCC-gatekeeper.

Contention-based forwarding (CBF) is one of the standardized multi-hop forwarding algorithms in the ETSI protocol stack. It is a receiver based forwarding algorithm, which works with timers and overhearing. DCC-gatekeeper adds additional queuing delay to the overall link layer delay seen by CBF, which destroys its overhearing functionality and causes a complete malfunctioning of CBF. To overcome this problem the DCC-gatekeeper has to be enhanced in order to maintain the overhearing.

So far existing studies on DCC only consider CAMs and single-hop packets [2] [3] [4] [5]; they do not address multi-hop forwarding and its related issues. Instead, our paper focuses on the networking aspects for multi-hop communication and the influence of DCC on the forwarding algorithm CBF as a state-of-the-art and standardized solution. The main

[1]IEEE 802.11 OCB = Outside of the context of the Basic Service Set, formerly known as IEEE 802.11p.

[2]ETSI standards are available at http://www.etsi.org/standards.

contributions of the paper are: first, we take the IEEE 802.11 standard MAC schemes DCF and EDCA as reference cases and study the performance of CBF when the default DCC (DCC Plain) is present. Second, we provide a solution for DCC, named DCC Advanced, to overcome the problem of the CBF's overhearing functionality at high link-layer delays. Third, we evaluate and compare the four protocol variants (DCF, EDCA, DCC Plain, DCC Advanced) with respect to reliability, latency, channel load, and CBF functionality.

The remainder of this paper is organized as follows. Sec. II provides the technical background for MAC, DCC and forwarding algorithms, followed by the description of our proposed scheme in Sec. III. Sec. IV defines the evaluation scenario and metrics, Sec. V presents the performance results and Sec. VI draws conclusion.

II. Base Algorithms for CBF, MAC and DCC

This section introduces the relevant components for medium access and decentralized congestion control.

Contention-based Forwarding (CBF) is a forwarding algorithm for multi-hop distribution of packets in ad hoc networks [9]. With CBF, a node always broadcasts a data packet to all neighbors and these neighbors compete to become the forwarder of the packet. When the optimal forwarder is found, the others refrain from forwarding. Opposed to conventional forwarding algorithms, where the sender of a packet determines the forwarder, CBF is a receiver-based scheme. In a previous study we found that CBF among several other standardized algorithms for VANETS[3] is the best algorithm with high reliability, robustness and latency [10].

The sole requirement of CBF is the fact that the nodes need to know their ego-geographical position and the position of the sender; the first can be acquired by GNSS, the latter is carried in the packet header. Upon reception of a packet, a node buffers it and starts a timer. The timer duration depends on the own geographic position and the sender position where the packet with the largest progress towards the destination (unicast) or distance to the sender (broadcast) has the shortest timer. The maximum timer length is 100 ms. When the timer expires the packet is taken from the buffer, passed to the MAC, and re-broadcasted when the channel access is granted. This transmission is overheard by the neighbor nodes. If they receive the packet a second time, they stop their timers and remove it from their buffer. In order to recognize if a duplicate of a packet arrives, nodes maintains a packet duplication list that logs the identifier (ID) of the received packet (combination of source node identifier and sequence number). A flow diagram of CBF is provided in Fig. 2.

The CBF algorithm works as long as the duration of time between passing the packet to the MAC and reception of the packet at other nodes (link-layer delay) is small, at least one dimension smaller than the CBF timer duration. When the link-layer delay gets larger, typically because of the unrestricted medium access delay of CSMA/CA, the queuing time of a packet in the MAC queue can exceed the maximum CBF timeout. In this case, overhearing fails [10] and the efficient forwarding turns into an inefficient flooding of the network (sometimes also referred to as broadcast storm).

[3]ETSI EN 302 636-4-1 V1.2.1

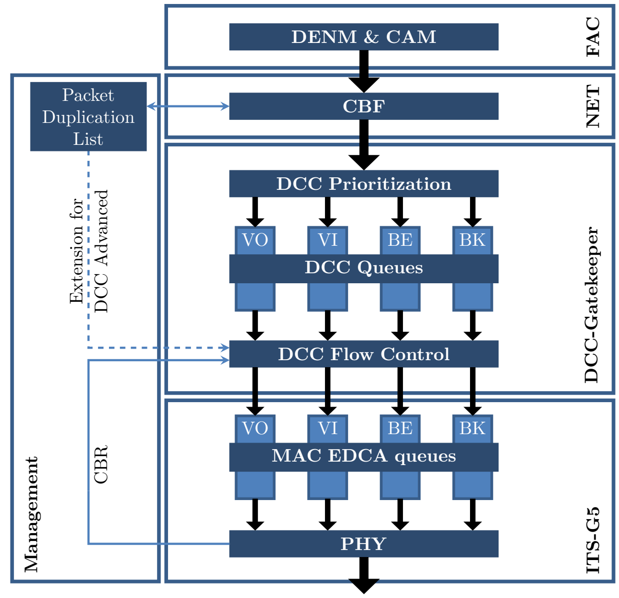

Fig. 1. Protocol stack for ITS-G5 based communication illustrating the interaction among the different protocol layers: the DCC-gatekeeper between the NET and MAC layer reduces the packet rate. The difference between DCC Plain and DCC Advanced lies in the exchange of information about duplicate packets among the layers. *CBR* indicates the channel busy ratio, provided by the PHY to the DCC Flow Control.

TABLE I. Parameters for ITS-G5 for CW and AIFS[5]. AC_VO has the highest priority.

AC	**CW_{min}**	**CW_{max}**	**AIFS** (Slots | Time)	**Messages**
AC_VO	3	7	2 | 58 μs	High-priority DENM
AC_VI	7	15	3 | 71 μs	DENM
AC_BE	15	1023	6 | 110 μs	CAM
AC_BK	15	1023	9 | 149 μs	Others

Distributed Coordination Function (DCF) is the basic MAC scheme of IEEE 802.11 and based on carrier sense multiple access with collision avoidance (CSMA/CA) [11]. CSMA/CA handles a single packet at a time, i.e. uses a single queue in which all packets are queued and a FIFO[4] queuing strategy. If the packet queue is full, newly arriving packets are discarded.

Enhanced Distributed Channel Access (EDCA) extends DCF and introduces several access classes (AC) to differentiate the channel access to different priorities. The differentiation is realized by several independent MAC entities per node with different configurations for inter-frame spaces (AIFS) and contention windows (CW) (see Tab. I) [12]. CAMs and DENMs are buffered in the AC_BE and AC_VI queue, respectively.

DCC Plain is the DCC mechanism with the DCC-gatekeeper functionality, where the LIMERIC algorithm is used to determine the DCC-gatekeeper's packet rate. The DCC-gatekeeper is explained next. We regard this as a reference case for comparison.

DCC-Gatekeeper is a component of decentralized congestion control (DCC). DCC aims at keeping the channel load below a predefined threshold in order to prioritize the data traffic and ensures that high priority safety messages are transmitted.

[4]FIFO – First In First Out.

[5]ETSI EN 302 663 V1.2.0, ETSI TS 102 636-4-2 V1.1.1

For this reason the DCC-gatekeeper is introduced, located below the network layer and above the EDCA-based MAC (see Fig. 1).[6] The DCC-gatekeeper prioritizes the messages according to their traffic class and controls the transmission intervals. It consists of three main components, which will be explained next.

DCC Prioritization differentiates the packets coming from the upper layer according to their traffic class and assigns them to the corresponding queue, e.g. CAMs and DENMs are buffered in the AC_BE and AC_VI queues, respectively. The traffic classes are mapped to the access classes from the EDCA queues with four priorities.

DCC Queues are on top of the EDCA queues with the same priorities. In contrast to EDCA the DCC flow control is in charge of the de-queuing process and the queues are dependent, which means that the queue with the highest priority is dequeued first.

DCC Flow Control de-queues packets from the DCC queues and passes them to the corresponding EDCA queue (see Fig. 1), whereby we assume a one-to-one mapping between DCC and EDCA queues. The dependency and prioritization among the DCC queues can lead to the effect that packets 'starve' in lower priority queues when the channel is saturated and only higher prioritized packets are sent. DCC flow control inserts a delay T_{off} while the DCC-gatekeeper is closed and subsequent packets are buffered in the DCC queues. When T_{off} expires the next packet is passed to the EDCA queues. The intention is to keep the channel load below a threshold. With growing channel load T_{off} is increased and fewer packets are transmitted, which leads to a decreasing channel load.

LIMERIC For the computation of the delay T_{off} we choose the rate control algorithm LIMERIC [2] and modify the algorithm in order to use the channel busy ratio CBR, provided by the PHY (see Fig. 1), as adjustment value. The algorithm determines node i's next packet rate $r_i(t_{k+1})$, which is the inverse of T_{off} (see Eq. (1), (2)). It works recursively, calculates the tracking error between the actual CBR and the CBR threshold CBR_{th} and takes this deviation into account for the computation of the next message rate. The rate converges to the value where CBR meets CBR_{th}.

$$r_i(t_{k+1}) = (1-\alpha)r_i(t_k) + \frac{\beta}{T_{\text{on}}}\left(CBR_{\text{th}} - CBR_i(t_k)\right), \quad (1)$$

$$T_{\text{off},i} = \frac{1}{r_i(t_{k+1})}. \quad (2)$$

In Eq. (1), $r_i(t_k)$ is node i's current packet rate. $CBR_i(t_k)$ indicates ith current channel busy ratio. α and β are constants that control the exponential forgetting and scaling, respectively. T_{on} is the packet air time. In Eq. (2), $T_{\text{off},i}$ is the delay until DCC flow control of node i fetches one packet from the DCC queue. The parameter values are listed in Tab. II. T_{off} is restricted to values between 10 and 1,000 ms, which avoids that the interval decreases to 0 with lightly loaded channels and grows to infinity in saturated channels. For details on LIMERIC, e.g. convergence behavior and stability criterion, we refer to [2].

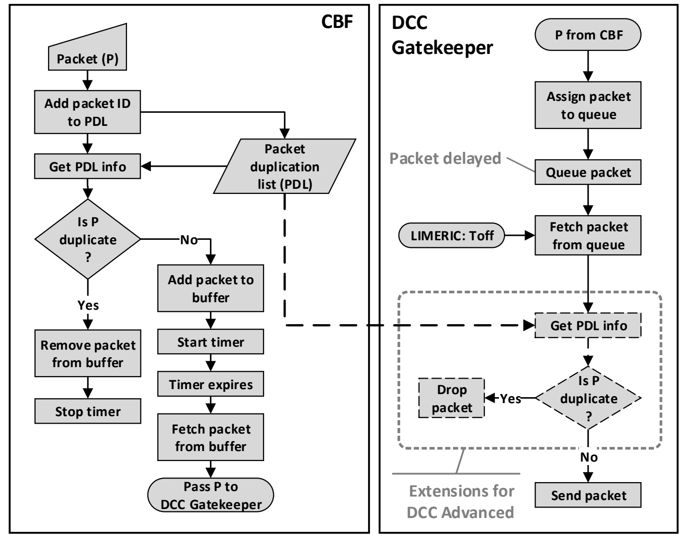

Fig. 2. CBF and DCC-gatekeeper flow diagram. The diagram shows the flow of a packet through the network layer and the DCC-gatekeeper extension.

III. DCC ADVANCED

DCC Advanced extends DCC Plain by enabling the DCC flow control to access the packet duplication list of CBF (see the dashed lines in Figs. 1 and 2). When the DCC flow control component de-queues a packet, it acquires the information about the existence of a duplicate packet from the packet duplication list. If the packet is indeed a duplicate, the DCC flow control denies further processing of the packet and thereby avoids erroneously forwarding (see the dashed box in Fig. 2).

The DCC-gatekeeper stores packets in the DCC queues until the DCC flow control processes them for transmission. The interval at which packets are fetched from the DCC queues is T_{off} and computed by the LIMERIC algorithm. The delay caused by the DCC queues enlarges the medium access time by a magnitude of 100 compared to the CSMA/CA access time. This long delay disables the overhearing functionality of CBF: when a packet is waiting in the DCC-queues, the CBF timers in neighboring nodes may expire and their packets are passed to the DCC-gatekeeper. Since in DCC Plain there is no possibility to prevent packets from being sent once they are forwarded by CBF, the DCC-gatekeeper passes all packets to the EDCA queues. As a result, the overhearing in CBF fails and packets are unnecessarily forwarded. In order to solve this problem we try to utilize the the information whether a packet copy arrives at a node while the packet is already stored in the DCC-queues.

For CBF a packet duplication list is already available. This list is used to log all arriving packets by means of the packet's source ID and sequence number, and prevents packets from being sent if a packet copy during the CBF timeout period arrives. In DCC Advanced, the DCC flow control has access to the packet duplication list and if a packet is received twice, both packets are discarded (see dashed box in Fig. 2); in this way the overhearing functionality of CBF is restored.

[6]ETSI TR 101 612 V1.1.1

The advantage of our proposal is the fact that the CBF functionality in the presence of high link-layer delay is restored, such that CBF is able to control the number of packet re-transmissions. Hence, in scenarios with many nodes and high data load, we get a smaller number of sent packets and the lower CBR results eventually in a higher reliability. Alternative approaches may try to modify the CBF algorithm to cope with large link-layer delays, such as enhancing the CBF timeout by the queuing time in the DCC queues. However, we argue that such modifications would increase the complexity of the CBF algorithm. Our approach does not modify the CBF, but comes at the costs of a more challenging cross-layer interaction: The DCC flow control needs to request the information whether a duplicate packet exists and to wait for the reply.

IV. Evaluation Scenario and Metrics

For the performance evaluation, we consider a typical freeway scenario with three lanes per direction and a speed limit of 120 km/h, i.e. the maximum permitted velocity that vehicles are allowed to travel. The vehicles are uniformly distributed and the density of nodes varies between 5 to $60\,\mathrm{nodes/km/lane}$, covering a range from free flow to traffic jam. Reasoning for selection of the vehicle densities can be found in [10]. To avoid edge effects at the end of a freeway, the simulated freeway forms a large circle with a diameter of 6 km. We further assume that all vehicles are equipped with the communication technology. The simulated nodes implement the ITS-G5 PHY and MAC, GeoNetworking with CBF, DCC with the DCC-gatekeeper, and the facility layer protocols CAM and DENM, all following the respective ETSI standards EN+302 663 V1.2.0, EN 302 636-4-1 V1.2.1, TS 102 636-4-2 V1.1.1, TS 103 175 V1.1.1 and EN 302 637-3 V1.2.2.

The simulation is executed in three steps: in the first 5 s of the simulation only CAMs are generated. After five seconds, 500 nodes start transmitting DENMs with an interval of 1 s for 90 s, which corresponds to a total number of 45,000 sent messages. The overall simulation runs for 100 s. The nodes stop generating new DENMs 5 s before the end of the simulation to allow the remaining DENMs to reach their destination. Therefore, the simulation has transient phases at the beginning and the end of a simulation run. A simulation is executed for the four protocol options, i.e. DCF, EDCA, DCC Plain and DCC Advanced. CBR_{th} is set to 0.65, which is proposed in ETSI TR 101 612 V1.1.1. All queues in the simulated setup have a maximum queue size of 400 packets and a maximum queuing time of 10 s each.

The simulation was performed with the network simulator ns-3[7], version 3.23. We have extended the basic simulator by protocols for GeoNetworking, CAM, DENM, and the DCC-gatekeeper. The mobility of the vehicles were controlled by traces that were generated with the road traffic simulator SUMO[8]. The simulation is performed with up to 6,500 nodes where, because of the very high number of nodes in the simulation setting, the change of the seed did not alter the results. Therefore only a single seed was taken. A summary of all simulation parameters can be found in Tab. II.

[7]http://www.nsnam.org

[8]SUMO – Simulation of Urban MObility.

TABLE II. Simulation Parameters

Parameter	Values
Vehicle density	5-60 veh/km/lane (18.3 km, 3+3 lanes)
PHY and MAC protocols	ITS-G5 (IEEE 802.11p)
Data rate	$6\,\mathrm{Mbit/s}$
Tx Power & antenna gain	20 dBm + 9.1 dB
Channel bandwidth	10 MHz at 5.9 GHz
Path loss model	ThreeLogDistance: channel exp. 1.9 - 3.8
Fading model	Nakagami - shape factor 1
Energy detection threshold	-95 dBm
Carrier sense threshold	-99 dBm
Max - queue size | - queuing time	400 packets | 10 s
Simulation duration	100 s
CAM interval | size	0.5 s | 512 bytes
DENM interval | size	1 s | 512 bytes
Total number of sent DENMs	45,000 (500 vehicles, every sec. for 90 s)
Geo-area	1,500×100 m behind the vehicle
LIMERIC: α | β | T_{on}	0.001 | $1/2000$ | $872\,\mu$s
CBR threshold – CBR_{th}	0.65
$T_{\text{off,min}}$ | $T_{\text{off,max}}$	10 ms | 1000 ms

We define the following evaluation metrics to compare and evaluate the algorithms in terms of reliability, latency and channel load aspects.

Inverse Node Coverage Ratio (Inverse NCR): Many safety and traffic efficiency applications, such as traffic jam ahead warning or emergency electronic brake light, need to disseminate information to vehicles in their vicinity. For this reason the application defines a relevance area. The relevance area is mapped to a geographical area, in which the communication system distributes the packet by means of multi-hop communication. Reliable communication implies that all nodes that are located inside the area receive the message at least once. The inverse NCR measures if a vehicle misses or drops the packet. For example, if 10 ve hicles are located inside the area and two nodes miss the message, the inverse NCR would be 20 %.

Average End-to-End Delay ($E2ED$): Is the duration of time between transmitting DENMs at the source until they arrive at the destination, averaged over all messages.

$$E2ED = \frac{1}{N}\sum_{i=1}^{N}\left(t_{\mathrm{Rx},i} - t_{\mathrm{Tx},i}\right), \tag{3}$$

where N is the number of received messages in the geo-area, $t_{\mathrm{Tx},i}$ and $t_{\mathrm{Rx},i}$ are the timestamps of the generated and received DENMs, respectively. The $E2ED$ is measured from the source to the destination including all forwarders.

Channel Busy Ratio (CBR): The CBR describes the data load on a wireless channel as the ratio between the fraction of time the channel is detected to be busy, i.e., a packet transmission is ongoing, and the fraction of time a channel is idle (no transmission is ongoing), for a pre-defined time interval. For example, a CBR of 0.8 means that for 80 % of the time the channel is busy.

Data Traffic Overhead (DTO): The data traffic overhead exhibits the number of sent frames at the PHY divided by the number of sent DENMs at the facility layer. Compared to CBR, the DTO considers the amount of data whereas CBR only measures the channel state, and does not count the number of generated messages.

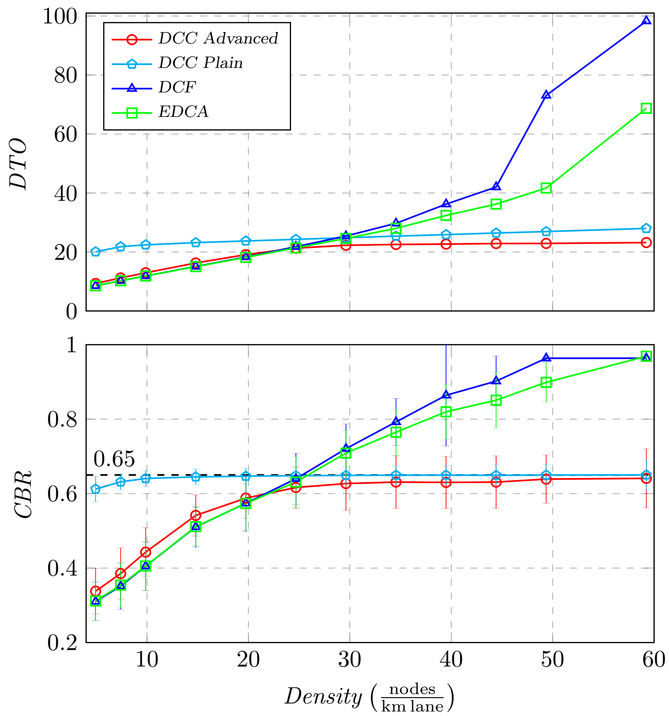

Fig. 3. Average channel busy ratio CBR (bottom) and data traffic overhead DTO (top) over node density. DCF and EDCA reach a CBR of 1 which means that the channel is always busy. DCC Plain and DCC Advanced converge to the predefined CBR threshold of 0.65.

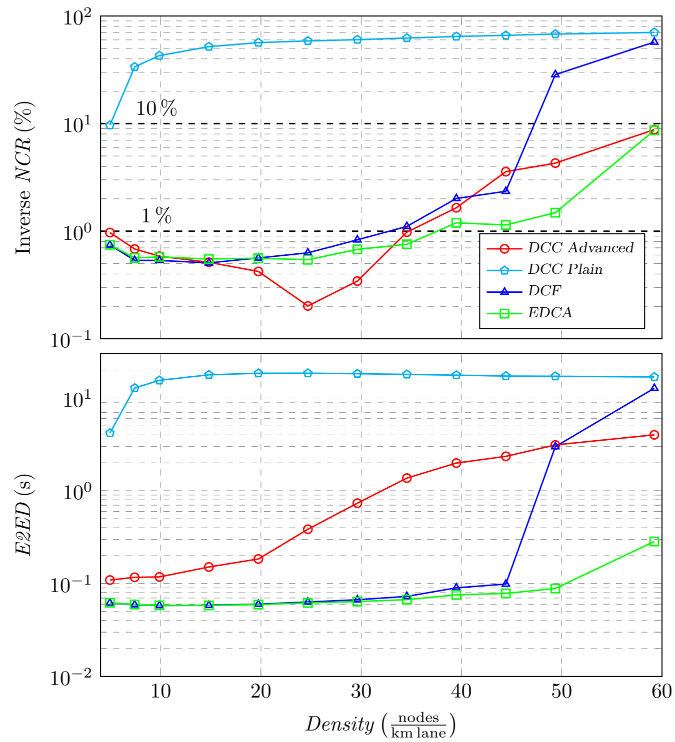

Fig. 4. Inverse node coverage ratio (top) and average end-to-end delay (bottom). The inverse node coverage ratio represents the ratio of nodes in a predefined geographic area which does not receive a packet, e.g. if the graphs are below the dotted $1\,\%$ line, more than $99\,\%$ of nodes received the packet.

V. Performance Evaluation

This section describes the simulation results in terms of channel load, reliability and latency aspects.

A. Channel Load

Figure 3 shows the data traffic overhead (top) and the average channel busy ratio (bottom). For low densities, DCC Plain utilizes the channel twice as much as the other algorithms and quickly reaches the CBR threshold of 0.65. This behavior can be explained by the malfunctioning CBF mechanism. For higher node densities the DTO and CBR for DCF and EDCA increase due to the growing number of sent packets. DCC Advanced has the same behavior as DCF and EDCA up to a node density of $25\,\text{nodes/km/lane}$. From this point, the LIMERIC algorithm increases T_{off} and the DCC-gatekeeper reduces the packet rate. For the whole range the average CBR for DCC Plain and DCC Advanced is below the predefined channel busy ratio threshold of 0.65, and the DTO remains constant up to high node densities[9]. It is worth noting that DCF and EDCA reach a CBR of almost 1, which means that the channel is busy at any time. In this situation it is not possible to ensure that a, for instance, high priority message can be sent and received.

B. Reliability and Latency

Figure 4 exhibits the inverse node coverage ratio (top) and the end-to-end delay (bottom) over the node density, which measure the communication reliability and latency, respectively. The inverse NCR for DCC Plain exceeds $10\,\%$ for the lowest node density and increases rapidly to $60\,\%$ for $20\,\text{nodes/km/lane}$ due to the failing CBF algorithm. The reason is that all nodes attempt to forward all received packets, which causes a highly loaded channel (channel usage is at the permitted threshold of 0.65, see Fig. 3) and an increase in the DCC-gatekeeper queue size. In this queue the packets wait for channel access, which is not granted due to the congested channel, and therefore a huge amount of packets is discarded once they exceed the queuing lifetime. An additional effect of the long queuing time is that, when channel access is granted, all packets which have exceeded the queuing lifetime are dropped, and the packets which are shortly before their expiration time are forwarded. This causes a very high end-to-end delay of almost $20\,\text{s}$ at $15\,\text{nodes/km/lane}$.

Compared to DCC Plain, the other algorithms show a better behavior and up to $35\,\text{nodes/km/lane}$ they are below an inverse NCR of $1\,\%$. When DCF reaches an $E2ED$ of $100\,\text{ms}$, the CBF mechanism using DCF fails, and the inverse NCR increases up to $60\,\%$, whereby EDCA remains below $10\,\%$ also for highest node densities. The explanation for the better performance of EDCA is that CAMs and DENMs are stored in different queues with different priorities where DENMs having a higher priority than CAMs. The analysis of the DENM's EDCA queue figured out (not shown in this paper) that the queuing time for the majority of DENMs is less than $100\,\text{ms}$ even for high node densities, which results in a working overhearing mechanism for CBF and much less DTO for high node densities (see Fig. 3, top).

[9] Certainly, single nodes exceed CBR_{th} and the LIMERIC algorithm has to react but in average CBR sticks to the threshold.

EDCA and DCC Advanced present the best behavior in terms of the inverse NCR, whereas DCC Advanced is better in the density range from 15 to $35\,\mathrm{nodes/km/lane}$ and EDCA is slightly better in the density range from 35 to $50\,\mathrm{nodes/km/lane}$. The reason why DCC Advanced does not fail is that the DCC-gatekeeper flow control checks the packet duplication list. If a packet is received a second time, it is dropped. Thereby the overhearing works up to long queuing times.

The $E2ED$ for DCC Advanced is much longer compared to EDCA. The reason is the lower CBR threshold in DCC Advanced where just $65\,\%$ of the time the channel is used and the rest of the time the channel remains idle. The less channel usage is achieved by the control of the packet rate in the DCC-gatekeeper and therefore if less packets are sent the channel load decreases. The disadvantage is an increasing queue size and queuing time, which causes the long $E2ED$. Thus, we can state the smaller CBR is under the cost of a longer $E2ED$ with an unchanging NCR.

EDCA shows the best performance if we are just interested in a high NCR and short $E2ED$. But for a more comprehensive analysis and for safety applications for which the reliability and latency of message reception is a critical aspect, also the channel load has to be taken into account. From this perspective, DCC Advanced shows the best results: the inverse NCR is comparable or equal to EDCA with $35\,\%$ less channel usage. This comes at the costs of a longer $E2ED$ for the normal CAMs and DENMs. But if an application needs to send a high priority message in a hazardous situation the channel is in a suitable state with free resources. Then, high priority messages can be sent through the higher-priority DCC-gatekeeper VO-queue, which is empty and therefore no delay occurs.

VI. Conclusion

In this paper we have studied vehicular ad hoc networks based on IEEE 802.11 OCB, in particular the interaction among medium access control (MAC), decentralized congestion control (DCC) and contention-based forwarding (CBF). We have verified that for the standard MAC schemes DCF and EDCA the channel busy ratio (CBR) and the data traffic overhead (DTO), both measures for the load in the wireless channel, significantly grow with the vehicle density. We have further demonstrated that the proposed DCC-gatekeeper-based approach is able to effectively control the channel load (CBR and DTO). While CBF over DCF and EDCA shows good performance in terms of node coverage ratio (NCR) and end-to-end delay (E2ED), the use of the DCC-gatekeeper significantly lowers the CBF performance. Reason for the performance degradation is the fact that the DCC-gatekeeper queues packets with a delay that is in the same order of magnitude or longer compared to the maximum CBF timer duration. In this case, the overhearing mechanism of CBF fails and CBF turns into a simple, inefficient flooding scheme.

We have proposed a DCC extension, named DCC Advanced, that enables the DCC flow control to access the packet duplication list of CBF and restores CBF's overhearing functionality. Our simulations show that CBF with DCC Advanced considerably improves the NCR and E2ED compared to DCC Plain. Compared to DCF and EDCA (without any DCC), the E2ED is still worse, however, we expect that the presence of the DCC functionality leaves transmission resources for the possibility of instantaneously sent higher-priority messages. The performance evaluation of this effect – i.e. sending high-priority messages in addition to "normal" CAM and DENM data traffic in saturated channel conditions – is not investigated in this paper and is considered in future work.

Acknowledgment

This work has been performed in the framework of the FP7 project ICT-619555 RESCUE (Links-on-the-fly Technology for Robust, Efficient and Smart Communication in Unpredictable Environments), which is partly funded by the European Union. We also thank Jérôme Härri for fruitful discussions.

References

[1] A. Festag, "Cooperative Intelligent Transport Systems (C-ITS) Standards in Europe," *Communication Magazin*, vol. 12, no. 52, Dec. 2014.

[2] J. Kenney, G. Bansal, and C. Rohrs, "LIMERIC: A Linear Message Rate Control Algorithm for Vehicular DSRC Systems," *ACM VANET '11*, pp. 1–10, Sep. 2011.

[3] L. Le, R. Baldessari, P. Salvador, A. Festag, and W. Zhang, "Performance Evaluation of Beacon Congestion Control Algorithms for VANETs," in *IEEE GLOBECOM '11*, Dec 2011, pp. 1–6.

[4] M. Torrent-Moreno, J. Mittag, P. Santi, and H. Hartenstein, "Vehicle-to-Vehicle Communication: Fair Transmit Power Control for Safety-Critical Information," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 7, pp. 3684–3703, Sep. 2009.

[5] R. Stanica, E. Chaput, and A.-L. Beylot, "Physical Carrier Sense in Vehicular Ad-Hoc Networks," in *IEEE MASS '11*, Valencia, Spain, Oct. 2011, pp. 580–589.

[6] H. Song and H. S. Lee, "A Survey on how to Solve a Decentralized Congestion Control Problem for Periodic Beacon Broadcast in Vehicular Safety Communications," in *ICACT '13*, Jan. 2013, pp. 649–654.

[7] A. Autolitano, C. Campolo, A. Molinaro, R. Scopigno, and A. Vesco, "An Insight into Decentralized Congestion Control Techniques for VANETs from ETSI TS 102 687 V1.1.1," in *IFIP Wireless Days (WD '13)*, Valencia, Spain, Nov. 2013, pp. 1–6.

[8] G. Bansal, B. Cheng, A. Rostami, K. Sjoberg, J. B. Kenney, and M. Gruteser, "Comparing LIMERIC and DCC Approaches for VANET Channel Congestion Control," in *IEEE WiVeC '14*, Sep. 2014, pp. 1–7.

[9] H. Füßler, J. Widmer, M. Käsemann, M. Mauve, and H. Hartenstein, "Contention-based Forwarding for Mobile Ad Hoc Networks," *Ad Hoc Networks*, vol. 1, no. 4, pp. 351–369, Nov. 2003.

[10] S. Kühlmorgen, I. Llatser, A. Festag, and G. Fettweis, "Performance Evaluation of ETSI GeoNetworking for Vehicular Ad Hoc Networks," in *IEEE VTC Spring '15*, Glasgow, UK, May 2015, pp. 1–6.

[11] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, Mar. 2000.

[12] A. Brachman and L. Chrost, "On the Shortcoming of EDCA Queues in the Presence of Multiservice Traffic," in *Wireless and Mobile Networking*, ser. IFIP Advances in Information and Communication Technology, J. Wozniak, J. Konorski, R. Katulski, and A. Pach, Eds. Springer Berlin Heidelberg, 2009, vol. 308, pp. 69–79.