Distributed EDCA Bursting: Improving Cluster-based Communication in IVC

Michele Segata
University of Trento - DISI
msegata@disi.unitn.it

Davide Goss
University of Trento - DISI
davide.goss@studenti.unitn.it

Renato Lo Cigno
University of Trento - DISI
locigno@disi.unitn.it

ABSTRACT

Cluster-based communication is a staple topic in vehicular networks. Clustering communication nodes promise to reduce channel contention, enable building backbones and might improve spatial reuse. In this paper we propose a seemingly simple, yet unexplored idea: extending 802.11 frame bursting MAC access to multiple stations aggregated into a cluster. The focus of the paper is thus not building a cluster, but exploring what is the gain that can be achieved by the standard 802.11p channel access if we introduce the principle of frame bursting (presently not allowed in 802.11p standard, but the key factor for the efficiency of 802.11n/ac WiFi channel access). The key scientific question is if we can extend the frame bursting mechanism so that only the cluster leader contends for the channel reserving a Transmission Opportunity that is used by all the vehicles in the cluster transmitting a coordinated burst of frames. We describe in detail the idea, highlight the problems that can be encountered in its implementation (the devil, as usual is in details), and present some preliminary results for a special class of clusters: Cooperative driving platoons of cars.

CCS Concepts

•**Networks** → **Link-layer protocols; Mobile ad hoc networks;** *Network performance analysis;*

Keywords

Inter-Vehicle Communication; Clustering; EDCA Bursting, IEEE 802.11

1. INTRODUCTION

The evolution toward Intelligent Transportation Systems (ITS) requires Inter-Vehicle Communication (IVC), and indeed, communication will play a fundamental role in ITS implementation. Sharing information among vehicles to implement intelligent cooperation is a must to improve road safety and efficiency. IVC would enable hundreds of different vehicular applications (see [8, 13, 22] for an exhaustive list of examples) that will completely change not only our driving experience, but the entire transportation system.

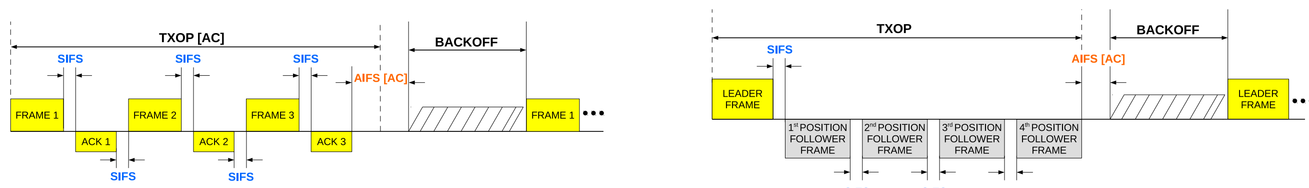
Among those, we have purely safety oriented applications, such as emergency braking and intersection collision avoidance [9, 21], where vehicles share their position, speed, and trajectory while approaching an intersection to forecast potential collisions and promptly warn the driver, or simply implement an optimal stopping strategy. Then we have a purely efficiency oriented application such as Virtual Traffic Light (VTL) [6, 7, 14], where standard traffic lights are substituted by a cooperative, self-organizing application that enables vehicles to automatically synchronize, reducing useless idle times. Finally, we also have mixed applications, such as platooning [5, 17, 25], that organizes vehicles in groups driven by an automatic system, minimizing inter-vehicle gaps (thus improving traffic flow) as well as the risk of collisions. Space forbids analyzing also the tens non ITS applications proposed for vehicular networks that require, or would benefit from, clustered communications.

In general, all IVC-based applications need to cope with a well known problem in wireless networks, i.e., channel congestion. Wireless LAN (WLAN) technologies such as IEEE 802.11p are known to suffer packet losses even at moderate channel loads and there are thus concerns on whether such technologies will be able to support the applications in heavily crowded scenarios. To cope with this problem, the research community proposed several channel congestion algorithms, i.e., protocols that adapt the transmission rate depending on some parameters (e.g., the current channel congestion state) to keep congestion under control and avoid packet losses (see [3, 4, 19, 24] to name a few).

All these approaches work on top of the MAC layer, trying to cope with its limitations. However, the IEEE 802.11 standard [2] already proposes some mechanisms that can help reducing congestion and improve the overall network throughput. Starting with the 802.11e amendment [1], the standard introduced several new features to support Quality of Service (QoS) in WLANs. Among those, we are interested in the frame bursting feature of the Enhanced Distributed Channel Access (EDCA). The idea of frame bursting is that a station does not contend for the channel for a single transmission, but for a certain amount of time defined as a Transmission Opportunity (TXOP). During this TXOP, a station is allowed to send multiple frames in a row, reducing protocol overhead and improving fairness among stations with different link qualities (see Section 2 for the detailed description of the mechanism).

IoV-VoI'16, July 05 2016, Paderborn, Germany

DOI: http://dx.doi.org/10.1145/2938681.2938686



(a) standard EDCA bursting

(b) proposed distributed EDCA bursting

Figure 1: EDCA bursting (as per 802.11e standard) and proposed Distributed EDCA Bursting (DEB).

In this work we propose a distributed implementation of the frame bursting feature, i.e., a station that wins a contention shares its TXOP with a group of vehicles, like a cluster. The idea is based on the observation that several applications explicitly or implicitly employ a clustering mechanism to share data among groups of vehicles. One example is intersection collision avoidance, where clusters of vehicles approaching the intersection naturally form on the road. Vehicles on one road needs to coordinate (e.g., determine which is the closer to the intersection), but also communicate and coordinate with vehicles on the other roads. Organizing the channel access per road-cluster can only improve efficiency and reduce channel contention. This is especially true when considering heterogeneous communication [12]. Another safety-related application is platooning: a platoon naturally forms a cluster of vehicles which share data for automatic control purposes. However, cluster-based communication in vehicular networks has received so much attention in the past [11] that there is no need for advocacy on its potential role in IVC.

In this paper we show the benefits of a distributed EDCA bursting approach for a specific use-case, i.e., a platooning application. The choice of platooning relies on the fact that this application creates, by definition, stable cluster of vehicles with a cluster-head, i.e., the leader. For space constraints, indeed, we disregard well-known issues connected to clustering, i.e., cluster formation, cluster-head selection, and stability of the clusters. These aspects will impact the performance of our approach, but in this study we simply focus on its feasibility. In particular, the contribution of this paper can be summarized as follows:

- We propose the distributed EDCA bursting mechanism, explaining in detail how standard EDCA bursting works and what is needed to implement that in a distributed fashion (Section 2);
- We analyze the performance of our approach for a platooning application, showing its benefits and its limitations compared to the classic channel access mechanism (Section 3);
- We list some research questions that still needs to be addressed before adopting this approach in real life applications (Section 4).

2. IEEE 802.11 BURSTING

The IEEE 802.11e amendment introduced several features to differentiate service, enhance throughput and improve fairness in WLANs. The Enhanced Distributed Channel Access (EDCA) distinguishes traffic mapping it to four different MAC queues, as opposed to the prior access scheme (named Distributed Coordination Function (DCF)), where all frames are managed by a single queue. Each EDCA queue has different MAC parameters, such as the amount of time spent in carrier sensing before transmitting or backing-off, the Contention Window (CW) size, maximum transmission duration, etc. With EDCA, regardless from the number of logical queues, a station contends the channel for a TXOP, and can send multiple frames as long as it holds the right to transmit. In general this feature is called *frame bursting*. Chipsets pre-802.11e instead contend for the channel to send a single frame: N frames require N contentions. This channel access mechanism results in an enormous amount of overhead due to the backoff procedure, and causes unfairness between stations using different link speeds. Imagine two stations with a link speed of 6 Mbit/s and 54 Mbit/s respectively, that need to send MPDUs of the same size. If they access the channel one after the other repeatedly, the measured application layer throughput will be the same for both, but it will be lower than 6 Mbit/s. By assigning the channel to the stations for a certain amount of time (a TXOP), instead, the two stations will fairly share the channel time, and their application layer throughput will depend only on their link quality and the amount of stations concurrently trying to access the channel.

The 802.11e standard proposes two ways for obtaining a TXOP. The first one is through standard EDCA contention: a station that wins a backoff contention obtains the TXOP. The second one is through the HCF Controlled Channel Access (HCCA), i.e., when the Access Point (AP) implements the Hybrid Coordination Function (HCF). In this mode, the AP divides the time into Contention Periods (CPs) (where stations use standard EDCA) and Contention Free Periods (CFPs) (where the AP assigns the channel to stations that requested for it). During the CFP, the AP polls single stations assigning TXOPs, which can be used by the stations to send multiple frames in a burst.

In both cases, the standard allows stations to perform bursting only in managed mode, i.e., when associated to an AP. Indeed, it is a duty of the AP to inform associated stations whether bursting is enabled and, in case, how long the TXOP is. Thus, strictly sticking to the standard, frame bursting cannot be used in vehicular networks, where there is no AP to coordinate the access. Figure 1a shows a graphical representation of a frame burst as per 802.11e standard.

2.1 Distributed EDCA Bursting

The idea we propose is non-standard compliant, but perfectly fits clustering scenarios. Take as an example a platooning application where, at each beacon interval, the leader vehicle and all its followers broadcast a packet to the vehicles in the same platoon. In here we consider a leader- and predecessor-following architecture, i.e., each vehicle expects to receive a beacon from the leader and from the vehicle in front [19]. The leader can act as a cluster head: when gaining access to the channel, it send a beacon that carries application information, but also reserves the channel for the amount of time required to send all the beacons of the followers by setting the Network Allocation Vector (NAV) of the MAC frame. Inside the beacon the leader includes, together with application layer data, the identifiers of the vehicles that should participate in the bursting procedure during the reserved TXOP: This implements a modified version of the CF-Poll frame of 802.11e. Upon reception of the leader beacon, the followers compute the identifier of the vehicle that should transmit immediately before each of them. The first vehicle in the list will schedule a transmission one Short Inter-Frame Space (SIFS) after the end of leader's frame. When receiving the frame of the first vehicle, the second in the list does the same, and the process continues until the last vehicle in the list. Figure 1b shows the working principle. Each vehicle properly sets the NAV to protect the remaining part of the leader reserved TXOP: This mechanism is useful to inform other platooning leaders if they failed to decode one or more beacons in the burst. In the case of a failed reception (e.g., the third vehicle is unable to decode second vehicle's frame), the distributed bursting stops.

Distributed EDCA bursting can reduce protocol overhead times caused by the backoff procedure, but it has some specific requirements. First of all, it requires a cross-layer approach: the application and the data-link layer must share information. The application is in charge of forming the cluster and decide which node is the cluster head. This information must be shared with the MAC. Moreover, the MAC needs to fetch data from the application in an "on-demand" fashion. When a node receives the frame of the previous vehicle in the list, it knows that it will need to send a beacon within a SIFS. The MAC queue, however, might either be empty or include an outdated packet. Thus a more effective, cross-layer technique would be to have the MAC fetch the most recent information just before sending the beacon.

3. PERFORMANCE EVALUATION FOR A PLATOONING APPLICATION

We test the performance of the proposed bursting mechanism (referred to as DEB) by means of simulations. We implement the protocol as an extension of the standard MAC layer included in the Veins framework [23], and we compare it against standard access mechanism (referred to as DCF). As anticipated, we consider a platooning scenario implemented in Plexe [20], thus we use the words platoon leader and cluster head, as well as platoon follower and cluster member, interchangeably. For both protocols we consider a beacon rate of 10 Hz, a standard value considered for a platooning application [15]. When using standard DCF, all vehicles send 10 beacons per second. With DEB, instead, only the cluster heads schedule period beacons, while the followers send their beacons according to the bursting scheme. The scenario reproduces a 4-lane freeway where 8-car platoons travel with a constant speed of 100 km/h. We consider a total number of cars going from 64 to 640 to investigate the behavior of the protocol under different network loads. Moreover, we consider two different transmit power settings for both DCF and DEB. In the first one (No TXPC) all vehicles use the

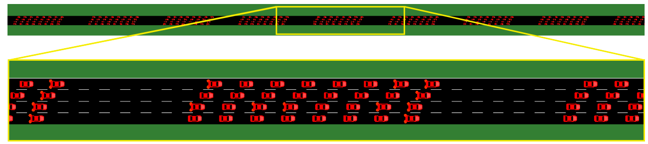


Figure 2: Screenshot of the simulation scenario.

Table 1: Network and road traffic simulation parameters.

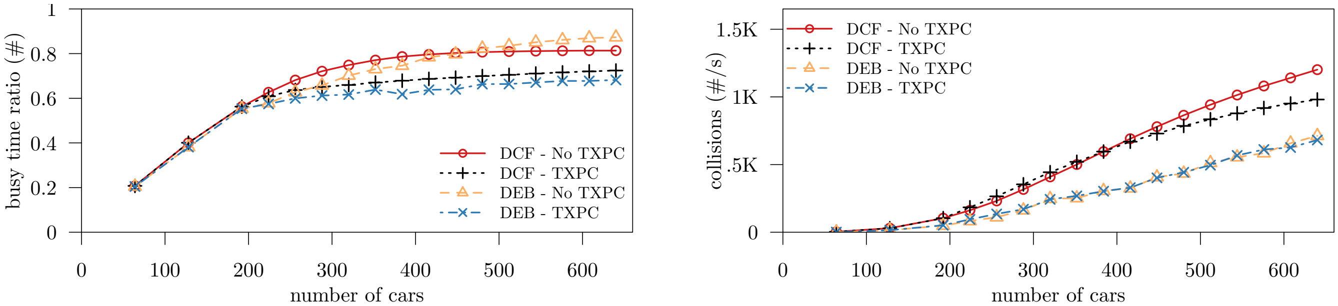
	Parameter	Value
communication	Path loss model	Free space ($\alpha = 2.0$)
	Frequency	5.89 GHz
	Bitrate	6 Mbit/s (QPSK R = 1/2)
	Transmit power	20 dBm and 0 dBm
	CCA threshold	−65 dBm
	Noise floor	−95 dBm
	Minimum sensitivity	−94 dBm
	PHY model	IEEE 802.11p
	MAC model	1609.4 single channel (CCH)
	Access category	AC_VI
	MSDU size	200 B
	Beacon rate	10 Hz
mobility	Speed	100 km/h
	Platoon size	8 cars
	Car's length	4 m
	Number of cars	64, 128, 192, 224, 256, 288 320, 352, 384, 416, 448, 480 512, 544, 576, 608, and 640
	Engine lag τ	0.5 s
	CACC's C_1, ω_n, ξ, d_d	0.5, 0.2 Hz, 1, 5 m
	ACC's T, λ	1.5 s, 0.1

same transmit power (20 dBm). In the second (TXPC) only the cluster heads (i.e., the leaders) use a transmit power of 20 dBm, while cluster members (i.e., the followers) use a reduced transmit power of 0 dBm. The following vehicles maintain a gap of 5 m to the one in front by using the California PATH's Cooperative Adaptive Cruise Control (CACC) [16], while leaders maintain an inter-platoon headway time of 1.5 s using a standard Adaptive Cruise Control (ACC). Figure 2 shows a screenshot of the simulation scenario, while Table 1 lists other specific simulation parameters (see [17] for details on the control parameters). To obtain a higher statistical confidence, we repeat each simulation setup 10 times.

3.1 General Networking Perspective

We begin the analysis by considering a general networking perspective, in particular by observing the channel busy ratio and the experienced frame collisions. The channel busy ratio indicates the amount of time the PHY layer senses the channel as busy. In the simulation, each vehicle samples its own channel busy ratio over one second intervals. Veins signals a collision on a frame when this can not be decoded due to interference. Each vehicle logs the number of frame collisions once per second.

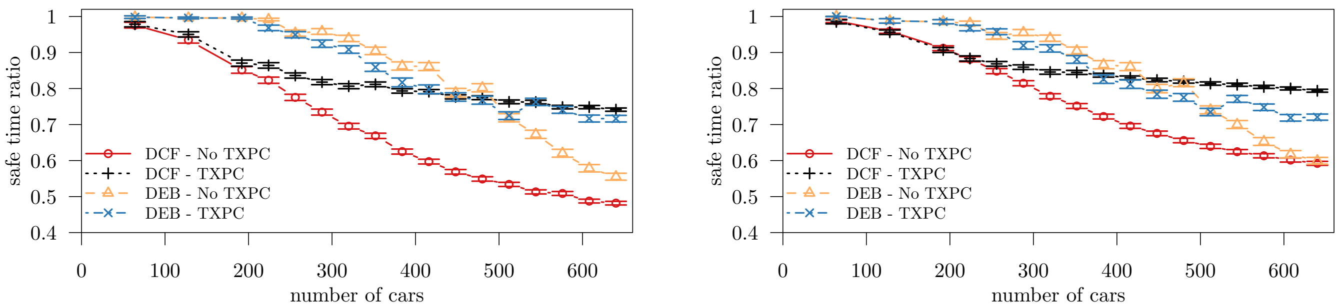
Figure 3a shows the average busy ratio over all cars and all repetitions for the four considered approaches. For a limited scenario size (i.e., up to roughly 200 vehicles) there is no notable difference between the four approaches. Above 200 vehicles, the two DCF protocols start to show different busy ratio values. When using TXPC the interference range is much more limited, thus the number of vehicles can increase without impacting on performance. Indeed, the busy ratio for DCF–TXPC slightly increases with the number of vehicles only because of the presence of new leaders, which transmit



(a) busy ratio

(b) experienced collisions

Figure 3: Comparison of DEB against standard DCF in terms of generic network metrics.



(a) leader packets

(b) front vehicle packets

Figure 4: Comparison of DEB against standard DCF from the application layer perspective: Safe time for a maximum allowable delay $\delta_{\mathrm{req}} = 100\,\mathrm{ms}$.

their beacons at full power. However, the protocol does not reach complete saturation. DCF–No TXPC, instead, causes the network to saturate, reaching the maximum possible busy ratio of 80 % at around 400 vehicles.

DEB–TXPC behaves very similarly to DCF–TXPC, but above 200 vehicles it uses roughly 5 % less channel resources, independently of the number of cars. DEB–No TXPC, compared to DCF–No TXPC instead, saves 5 % channel usage between 200 and 400 cars. At around 450 vehicles, the two lines crosses and, for 640 cars, DEB–No TXPC results in a higher channel utilization than DCF–No TXPC. This is caused by the higher channel saturation point for DEB: The lower protocol overhead due to the lower number of backoffs increases the amount of time the channel can be used for useful transmission.

Figure 3b shows the measured number of collisions per second, averaged over all vehicles and repetitions. Both DCF approaches have a similar behavior and cause a larger number of collisions per second compared to DEB. In particular, DCF approaches start to diverge around 450 cars, i.e., when the channel starts to saturate, coherently with what we show in Figure 3a. With the channel reservation mechanism provided by DEB, instead, collisions are less likely to occur. Both DEB approaches behave very similarly, and cut the collision rate by 50 % with respect to DCF–No TXPC.

3.2 Application Layer Perspective

The analysis in Section 3.1 only looks at the performance from a pure networking perspective. However, frames delivered by beaconing protocols are used by the applications on top of them, and for this reason it is important to understand whether application requirements are fulfilled. To this purpose, in [18] we define an application layer metric for a CACC named *safe time ratio*, which measures the amount of time a vehicle is in a safe state, i.e., when CACC requirements are met. We formally define the metric as follows. Let δ_{req} be the CACC requirement in terms of maximum allowable delay, and let $\mathcal{D}$ be the set of all inter-message delays recorded by a vehicle. The set $\mathcal{D}$ only includes the delay of either leader or front vehicle packets, as they are the ones required by the CACC application. The set of all delays satisfying the requirement δ_{req} is defined as

$$\mathcal{D}_{\mathrm{safe}} = \{d : d \in \mathcal{D} \wedge d \leq \delta_{\mathrm{req}}\}\,. \tag{1}$$

Finally, we define the safe time ratio metric r_{safe} (for leader or front vehicle beacons) as

$$r_{\mathrm{safe}} = \frac{\sum_{d_{\mathrm{s}} \in \mathcal{D}_{\mathrm{safe}}} d_{\mathrm{s}}}{\sum_{d \in \mathcal{D}} d}\,. \tag{2}$$

When $r_{\mathrm{safe}} = 1$, all frames have been received within the time constraint. Conversely, when $r_{\mathrm{safe}} = 0$, no frame had an inter-arrival time lower than δ_{req}.

In this work, we set $\delta_{\mathrm{req}} = 100\,\mathrm{ms}$, which corresponds to receiving all the 10 packets per second scheduled by the application. Figures 4a and 4b plot the safe time ratio for leader and front packets respectively, with a 95 % confidence interval. For what concerns leader beacons we can see that using DCF followers immediately starts to miss some of the leaders' beacons, reducing the fraction of safe time. On the other hand, DEB always keeps followers in safe state regarding the leader up to 200 vehicles. This is due to the reservation of the TXOP: In DEB only the leaders content for channel access, and they do it either when the channel is free, or at the end of the NAV declared by another cluster head. In DCF, instead, cluster heads must content for channel

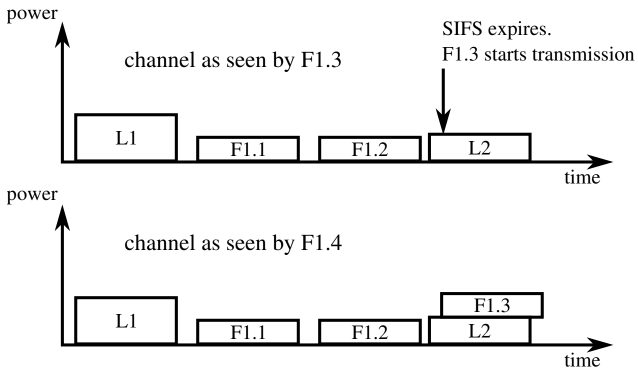


Figure 5: Distributed EDCA bursting problem in case of packet losses.

access together with cluster members, and this might easily lead to packet losses or delays even at moderate loads. For a higher number of vehicles, both DEB approaches start to loose performance up to 450 vehicles, where they "meet" DCF–TXPC. Above 450 vehicles, DEB–TXPC converges to the safe time ratio of DCF–TXPC, while DEB–No TXPC continues to loose performance. In any case DEB–No TXPC always improves application layer performance with respect to DCF–No TXPC. The same statement holds for DEB–TXPC and DCF–No TXPC.

For what concerns packets from the front vehicle (Figure 4b), the results are very similar. Up to 200 vehicles DEB approaches safely deliver all application-related packets. The performance then start to decrease up to 450 vehicles. There, both DEB approaches start to loose against DCF–TXPC. This is caused by a weakness in our basic protocol implementation, coupled with a simplified physical layer model implemented in the simulator. Imagine a cluster head sending a beacon reserving the channel for a TXOP. If a farther leader is unable to decode that frame it will not set its NAV, and might thus start transmitting while the other platoon is sending a burst. Figure 5 shows what can happen. Imagine to have the leader and the follower of platoon number 1 labeled L1, F1.1, F1.2, F1.3, F1.4, etc., respectively, and the leader of a farther platoon labeled L2. If L2 fails to decode L1's beacon, it might start to transmit between beacons in a burst, for example between F1.2 and F1.3 beacons, because followers use a reduced transmit power and they might not trigger L2's carrier sensing mechanism. If this happens, F1.3 and F1.4 synchronizes onto L2's frame, starting their reception process. When F1.3's SIFS expires, F1.3 starts sending the frame, but given that F1.4 is synchronized onto L2's frame, F1.3's frame is treated as noise and it is never received. In such a case, the bursting mechanism stops.

We believe the results in Figure 4b are an average between good and bad performance of vehicles at the front and at the back of a platoon, respectively. For cars at the back, indeed, the chance of not receiving the beacon from their front vehicle is higher, because the first error in the burst will stop the mechanism. A more realistic simulation model, however, might result in better performance because of the capture effect [10]. Modern WLAN devices can switch from the reception of a frame to another if the power level of the second one is higher than a certain threshold. The rationale is that, if the power of the second frame is much higher than the first, there is no chance of correctly decoding the first one, so a better option is to try to decode the second. If we consider the distance between two consecutive vehicles in our simulation (roughly 10 m) and a transmit power of 0 dBm, the signal power at the receiver should be around -67 dBm (considering a free space path loss with $\alpha = 2.0$ and a carrier frequency of 5.89 GHz). By setting a capture threshold of 10 dB, the NIC card would drop the frames of any leader farther than 300 m when a capture occurs.

Another possibility to prevent this is to switch off the autocorrelator during the SIFS, preventing the synchronization of the NIC onto other cluster head's frames. Moreover, we could implement a pre-scheduling mechanism, where the cluster members do not wait to receive the previous vehicle's frame to schedule the transmission, but they schedule it immediately after receiving the cluster head's frame. This way, loosing a beacon would not stop the bursting procedure.

4. CONCLUSION AND DISCUSSION

In this paper we presented a distributed EDCA bursting mechanism to improve cluster-based communication in IVC. The idea is to modify the standard 802.11e bursting mechanism, which usually works for a unicast communication between a station and the AP, and have a cluster head send a beacon to reserve the channel for the duration of a TXOP. The cluster head, with its beacon, polls cluster members, which send their data one after the other in a burst, i.e., having each of their frames separated by a SIFS.

We made a first implementation of the proposed mechanism in Plexe and tested its performance against the standard DCF-controlled channel access mechanism. We have shown that even this first prototype implementation has huge potential: It improves channel utilization by reducing protocol overhead and by increasing the channel saturation point. Moreover, it largely reduces the amount of collisions in the channel thanks to the reservation mechanism. Finally, we have also shown that this generic network performance improvement has a positive impact on application layer performance as well.

This first protocol proposal paves the road towards a more efficient data sharing protocol for intra-cluster communication in IVC, and opens a series of interesting research questions. First of all, we have seen that this first version might suffer reservation interruption by far-away cluster leaders that have not properly decoded the reservation beacons. The discussion of this phenomenon hints that in practice it might be negligible thanks to the capture effect of modern wireless transceivers. Part of our future work is thus to implement a more realistic physical layer model and prove this statement. Moreover, we would like to test other possible solutions to the problem, such as disabling autocorrelators during the SIFS or employing a pre-scheduling mechanism when receiving a beacon from the cluster head node.

Besides that, there are other interesting questions to address. For example, in a AP-managed 802.11 network it is the AP that dictates the maximum duration of a TXOP for all. In our case, each cluster-leader decides the TXOP duration: we have not yet studied the effect of different cluster dimensions on fairness and efficiency. An interesting question is about reliability. While standard 802.11 unicast frames are "protected" by acknowledgements, our approach is broadcast-based. Can we implement a reliable broadcast scheme to improve the protocol without adding excessive overhead (for example by including piggybacked acknowledgements)? Finally, in this paper we assumed stable, already formed clusters. Part of our future work is to consider other simulation settings where clusters form in a dynamic manner and their stability is influenced by vehicles' mobility.

We hope this paper can foster research on cluster-based frame bursting for IVC.

5. REFERENCES

[1] Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications - Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements. Std 802.11e-2005, IEEE, 2005.

[2] Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. Std 802.11-2012, IEEE, 2012.

[3] G. Bansal, J. Kenney, and C. Rohrs. LIMERIC: A Linear Adaptive Message Rate Algorithm for DSRC Congestion Control. *IEEE Transactions on Vehicular Technology*, 62(9):4182–4197, November 2013.

[4] European Telecommunications Standards Institute. Intelligent Transport Systems (ITS); Decentralized Congestion Control Mechanisms for Intelligent Transport Systems operating in the 5 GHz range; Access layer part. TS 102 687 V1.1.1, ETSI, July 2011.

[5] P. Fernandes. *Platooning of IVC-Enabled Autonomous Vehicles: Information and Positioning Management Algorithms, for High Traffic Capacity and Urban Mobility Improvement*. Phd thesis, University of Coimbra, Portugal, April 2013.

[6] M. Ferreira, R. Fernandes, H. Conceição, W. Viriyasitavat, and O. K. Tonguz. Self-organized Traffic Control. In *7th ACM International Workshop on Vehicular Internetworking (VANET 2010)*, pages 85–90, Chicago, IL, September 2010. ACM.

[7] F. Hagenauer, P. Baldemaier, F. Dressler, and C. Sommer. Advanced Leader Election for Virtual Traffic Lights. *ZTE Communications, Special Issue on VANET*, 12(1):11–16, March 2014.

[8] H. Hartenstein and K. Laberteaux, editors. *VANET - Vehicular Applications and Inter-Networking Technologies*. Intelligent Transport Systems. John Wiley & Sons, December 2010.

[9] S. Joerer, M. Segata, B. Bloessl, R. Lo Cigno, C. Sommer, and F. Dressler. A Vehicular Networking Perspective on Estimating Vehicle Collision Probability at Intersections. *IEEE Transactions on Vehicular Technology*, 63(4):1802–1812, May 2014.

[10] J. Lee, W. Kim, S.-J. Lee, D. Jo, J. Ryu, T. Kwon, and Y. Choi. An Experimental Study on the Capture Effect in 802.11a Networks. In *2nd ACM International Workshop on Wireless Network Testbeds, Experimental evaluation and Characterization (WiNTECH'07)*, pages 19–26, Montreal, Canada, September 2007. ACM.

[11] F. Li and Y. Wang. Routing in Vehicular Ad hoc Networks: A Survey. *IEEE Vehicular Technology Magazine*, 2(2):12–22, June 2007.

[12] A. Memedi, F. Hagenauer, F. Dressler, and C. Sommer. Cluster-based Transmit Power Control in Heterogeneous Vehicular Networks. In *7th IEEE Vehicular Networking Conference (VNC 2015)*, pages 60–63, Kyoto, Japan, December 2015. IEEE.

[13] National Highway Traffic Safety Administration. Vehicle Safety Communications Project Task 3 Final Report: Identify Intelligent Vehicle Safety Applications Enabled by DSRC. Project Report DOT HS 809 859, NHTSA, March 2005.

[14] T. Neudecker, N. An, O. K. Tonguz, T. Gaugel, and J. Mittag. Feasibility of Virtual Traffic Lights in Non-Line-of-Sight Environments. In *9th ACM International Workshop on Vehicular Internetworking (VANET 2012)*, pages 103–106, Low Wood Bay, Lake District, UK, June 2012. ACM.

[15] J. Ploeg, B. Scheepers, E. van Nunen, N. van de Wouw, and H. Nijmeijer. Design and Experimental Evaluation of Cooperative Adaptive Cruise Control. In *IEEE International Conference on Intelligent Transportation Systems (ITSC 2011)*, pages 260–265, Washington, DC, October 2011. IEEE.

[16] R. Rajamani, H.-S. Tan, B. K. Law, and W.-B. Zhang. Demonstration of Integrated Longitudinal and Lateral Control for the Operation of Automated Vehicles in Platoons. *IEEE Transactions on Control Systems Technology*, 8(4):695–708, July 2000.

[17] M. Segata. *Safe and Efficient Communication Protocols for Platooning Control*. Phd thesis (dissertation), University of Innsbruck, February 2016.

[18] M. Segata, B. Bloessl, S. Joerer, C. Sommer, M. Gerla, R. Lo Cigno, and F. Dressler. Towards Communication Strategies for Platooning: Simulative and Experimental Evaluation. *IEEE Transactions on Vehicular Technology*, 64(12):5411–5423, December 2015.

[19] M. Segata, F. Dressler, and R. Lo Cigno. Jerk Beaconing: A Dynamic Approach to Platooning. In *7th IEEE Vehicular Networking Conference (VNC 2015)*, pages 135–142, Kyoto, Japan, December 2015. IEEE.

[20] M. Segata, S. Joerer, B. Bloessl, C. Sommer, F. Dressler, and R. Lo Cigno. PLEXE: A Platooning Extension for Veins. In *6th IEEE Vehicular Networking Conference (VNC 2014)*, pages 53–60, Paderborn, Germany, December 2014. IEEE.

[21] M. Segata and R. Lo Cigno. Automatic Emergency Braking - Realistic Analysis of Car Dynamics and Network Performance. *IEEE Transactions on Vehicular Technology*, 62(9):4150–4161, October 2013.

[22] C. Sommer and F. Dressler. *Vehicular Networking*. Cambridge University Press, November 2014.

[23] C. Sommer, R. German, and F. Dressler. Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis. *IEEE Transactions on Mobile Computing*, 10(1):3–15, January 2011.

[24] C. Sommer, S. Joerer, M. Segata, O. K. Tonguz, R. Lo Cigno, and F. Dressler. How Shadowing Hurts Vehicular Communications and How Dynamic Beaconing Can Help. *IEEE Transactions on Mobile Computing*, 14(7):1411–1421, July 2015.

[25] V. Turri. *Fuel-efficient and Safe Heavy-duty Vehicle Platooning Through Look-ahead Control*. Phd thesis, KTH Royal Institute of Technology, October 2015.