Technical Report N° 462

Evaluation of Synchronous and Asynchronous Reactive Distributed Congestion Control Algorithms for the ITS G5 Vehicular Systems

Oyunchimeg SHAGDAR

Oyunchimeg.Shagdar@inria.fr

RITS Team, Inria Paris-Rocquencourt

Domaine de Voluceau, B.P. 105

78153, Le Chesnay, FRANCE

Inria
INVENTEURS DU MONDE NUMÉRIQUE

1 Introduction

The current work is conducted at INRIA (**French Institute for Research in Computer Science and Automation**), using the simulation tool NS3 (network simulator) and SUMO (Simulation of urban mobility.

A number of DCC algorithms such as Reactive DCC [1], LIMERIC [2] and AIMD adaptive control [3] have been proposed. The key differences lie in their ways of controlling the communication parameters. Having the CAM generation rate as the control parameter, the reactive DCC controls the rate following a parameter table; LIMERIC controls the rate based on a linear adaptive algorithm, and AIMD algorithm control the rate in a similar manner as TCP.

On the other hand, channel busyness ratio (CBR), which is the ratio of the time the channel perceived as busy to the monitoring interval, is the commonly agreed metric used to characterize channel load. Since the wireless channel is shared by the ITS-S that are in the vicinity of each other, CBR monitored at such ITS-Ss take similar values. As a consequence, the ITS-Ss may take synchronized reactions to the channel load, e.g., the ITS-Ss reduce/increase the transmission rate at around the same time. The first contribution of this work is thus to study such a synchronized DCC behaviour observed in reactive DCC algorithm. We pay an attention on the following different possible reactions of the CAM generator, which is responsible for adjusting the message generation rate as a means of DCC.

- *Timer handling*: In general, a transmission of a CAM is triggered by a timer, which is set to the CAM interval. Hence, upon being informed with a new CBR value (at an arbitrary point of time), the CAM generator may i) wait the expiration of the on-going timer and set the timer to the new CAM interval or ii) cancel the on-going timer and set it to the new CAM interval. We call the former and latter behaviours as *Wait-and-Go* and *Cancel-and-Go*.
- *Interval setting*: As mentioned above, CBR measured for the shared channel may lead to the situation where the nearby ITS-Ss increase/decrease the CAM interval at around the same time. This is especially true for the reactive DCC algorithm, which controls the rate following a table. Therefore, one can think of avoiding such a synchronized behaviours by applying random intervals. Hence, we can imagine 2 possible behaviours: upon reception of a new CBR value, the CAM generator sets the message generation interval to i) the value (say *new_CAM_interval*) provided by the table or ii) a random value (e.g., taken from the range [0, *new_CAM_interval*]) for the first packet and then follows the table. We call the former and latter behaviours as *Synchronized* and *Unsynchronized*.

Considering the above-mentioned behaviours of the CAM generator, we obtain the following 4 different versions of Reactive DCC:

- DccReactive-1: *Wait-and-Go* & *Synchronized*
- DccReactive-2: *Cancel-and-Go* & *Synchronized*
- DccReactive-3: *Wait-and-Go* & *Unsynchronized*
- DccReactive-4: *Cancel-and-Go* & *Unsynchronized*

The first contribution of this work is hence, to study and compare the performances of these different versions of reactive DCC to understand the synchronization issue and the underlying reasons.

Second contribution of this work is to have a close look to channel load characterization. While it is commonly agreed that CBR should be monitored over a certain interval (e.g., 100 ms), it is not clear if channel load should be characterized with the current value of CBR or it should also consider the past CBR values. To study this aspect, we define channel load (CL) as follows.

$$CL_n = (1-\alpha)\times CL_{n-1} + \alpha \times CBR_n \quad (1)$$

Here, CBR_n is CBR measured at the nth monitoring interval, CL_n is the channel load calculated upon measurement of CBR_n. As can be seen in (1), the weight factor, $\boldsymbol{\alpha}$, plays the key role for defining whether the channel load should consider only the last CBR or should pay an attention on its history. Obviously by choosing $\boldsymbol{\alpha}$=1, channel load is characterized by the "current" channel condition. In our study, we evaluate the performances of reactive DCC for different values of $\boldsymbol{\alpha}$.

The third contribution of this work is to study the DCC performance in road systems, which consist of ITS-Ss with different levels of sensing capability. Specifically, we consider that ITS-Ss sense the wireless channel at different levels, and as consequence, they perceive CL differently and react differently. To realise this study, ITS-Ss in the simulations are provided with random sensitivity offset values in the range of [-6, +6] dBm.

To summarize, the contributions of this work are as follows

- Contribution 1: Study on synchronization issue of DCC control.
- Contribution 2: Study on channel load characterization.
- Contribution 3: Study on non-identical sensing capabilities.

2 Used simulation tool(s)

The work is carried out using the open discrete event simulation environment NS3 (version 3.21) [4], and the traffic simulator SUMO (version 0.22) [5]. The key simulation modules, which are relevant to this work, are illustrated in Figure 1, where the modules written in red are newly developed software.

2.1 NS3

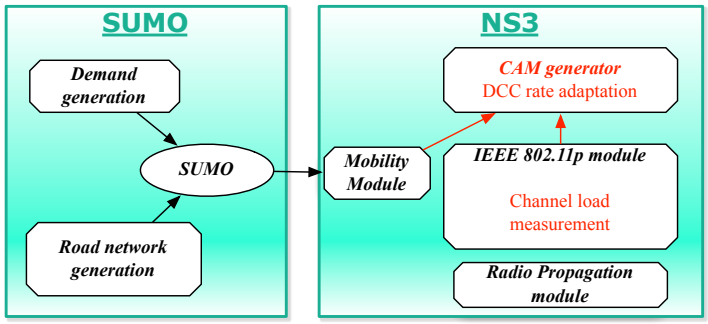

Figure 1 Simulators and the key modules relevant to the work

The latest stable version of NS3, NS-3.21, is used in this work. Among a number of new functionalities, it includes the WAVE system, which has the IEEE 802.11p (ITS-G5). The system follows the TCP/IP communication architecture. The key software components used in our simulations are a CAM generator, UDP/IP, IEEE 802.11p, radio propagation, and mobility modules.

The CAM generator is a newly developed module, which takes position and mobility information from the mobility module and periodically generates CAMs. The module is implemented with DCC rate adaptation algorithms. This work focuses on the reactive DCC algorithm. When the reactive DCC module is provided with a CL value (see (1)), it adjusts the CAM generation interval following the parameter table.

Messages generated at the CAM generator processed by the UDP and IP modules, and received at the IEEE 802.11p MAC. While, BTP/ GeoNetworking protocols are standardized in ETSI, utilizing UDP/IP is equivalent to utilizing BTP/Geonetworking for the objective of studying channel congestion caused by 1-hop broadcast messages (CAM). It should be noted that since the header lengths of UDP/IP and BTP/Geonetworking are different, the necessary message length adjustment is made at the CAM generator such that the length of the frames transmitted on the wireless channel have the same length to the case using BTP/Geonetworking.

The PHY layer of NS3 is extended with a CBR monitoring functionality, which monitors the channel activities and calculates CL. Since NS3 is an event-based simulator, the CBR monitoring module exploits the event notifications installed in NS3. In addition, the module holds a timer and calculates CBR in every $T_{monitor}$ interval following (2). It should be mentioned that timer setting is made independently at each ITS-S, and hence the CL notifications to the CAM generator is not synchronized among the individual ITS-Ss.

$$CBR = \frac{\sum T_{busy}}{T_{monitor}} \quad (2)$$

NS3 mobility module is responsible for mobility of ITS-Ss, and it is the interface of NS3 with the SUMO traffic simulator.

2.2 SUMO

The SUMO traffic simulator is used to generate road network and traffic following user-specified scenarios. The outputs of the traffic simulator are converted in a file format readable by the mobility module of the NS3 simulator.

3 Simulator configuration

Unless otherwise noted, the communication and road parameters take the values listed in Table 1.

Table 1 Common simulation parameters

Parameters	Value
Communication	
CAM default Tx rate	10 Hz
CAM message size	400 Bytes
Tx Power	23 dBm
$ED^{threshold}$	-95 dBm
EDCA Queue / TC	1 DENM / 3CAM
Modulation scheme	QPSK ½ 6 Mbps
Antenna pattern	Omnidirectional, gain = 1dBi
Access technology	ITS G5A
ITS G5 Channel	CCA
Fading model	LogDistance, exponent 2
Road network	
Lane width	3 m
Lanes in-flow	3
Lanes contra-flow	3
DCC parameters	
CBR monitor interval ($T_{monitor}$)	100 ms
α (see (1))	1

The parameter table of the reactive DCC algorithm is shown in Table 2.

Table 2 DCC Reactive Parameters

States	CL(%)	T_{off}
Relaxed	$0\% \leqslant CL < 19\%$	60
Active_1	$19\% \leqslant CL < 27\%$	100
Active_2	$27\% \leqslant CL < 35\%$	180
Active_3	$35\% \leqslant CL < 43\%$	260
Active_4	$43\% \leqslant CL < 51\%$	340
Active_5	$51\% \leqslant CL < 59\%$	420
Restricted	$CL \geq 59\%$	460

3.1 Simulation scenarios

The work is carried out for homogenous highway scenarios. Table 3 lists the road configuration. As shown in the table and illustrated in Figure 2, RSUs are installed in every 100 m on the middle lane.

The scenario consists of sparse, medium, dense, and extreme density classes; the density parameters are listed in Table 4.

Table 3 Road configuration

Class	Inter-vehicle distance
Highway length	1000 m
Lanes/Directions	3 lanes / 2 directions
RSU inter-location	100 m
Vehicle size	2m x 5m

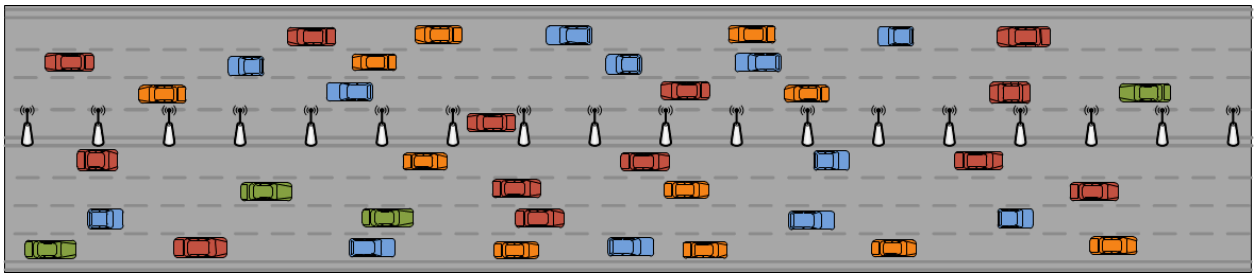

Figure 2 Illustration of a homogenous highway scenario.

Table 4 Density parameter for homogenous highway scenario.

Class	Inter-Vehicle distance	Mobility
Sparse	100 m inter-distance (3 lanes/ 2 directions)	Static/Mobile
Medium	45 m inter-distance (3 lanes / 2directions)	Static/Mobile
Dense	20 m inter-distance (3 lanes / 2directions)	Static/Mobile
Extreme	100 m inter-distance (3 lanes / 2directions)	Static

4 Simulation results

4.1 Introduction to the results

As described in Section 1.1, this work makes 3 contributions 1) study on synchronization issue of DCC control, 2) study on channel load characterization, and 3) study on non-identical sensing capabilities.

Following metrics are used for performance investigations.

- Packet delivery ratio (PDR): the ratio of the number of received packets over the number of transmitted (generated) packets. PDR is measured at individual ITS-S (vehicles and RSUs) targeting CAMs transmitted by each mobile ITS-S (i.e., vehicles).
- Packet Inter-Reception time (PIR): time gap between consecutive CAM messages. PIR is measured at individual ITS-Ss for received CAMs from each mobile ITS-S.
- Number of transmissions: the total number of CAM transmissions is counted for 20 milliseconds of time bins.
- CBR: the average CBR is calculated for 20 milliseconds of time bins.
- Jain's fairness index is calculated for the total number of transmissions from individual mobile ITS-Ss.

4.2 Contribution 1: Study on synchronization issue of DCC control

This subsection evaluates the performances of the four different versions of Reactive DCC: DccReactive-1, DccReactive-2, DccReactive-3, and DccReactive-4 for homogeneous static highway scenario. The performances of the mechanisms are compared against that of DccOff, which is the ITS-G5 system without distributed congestion control.

Figure 3 plots the average packet delivery ratio (PDR) of the reactive DCC mechanisms in contrast to that of DccOff. The horizontal axis is the distance between the receivers and the transmitters. First of all, we note that DccOff shows an excellent PDR performance in the sparse scenario (see Table 4), where obviously the channel is not congested. The channel congestion becomes an issue for medium, dense and extreme density classes, where PDR degrades down to 10% in DccOff. DccReactive mechanisms show better PDR than DccOff. The PDR improvement is much more significant for unsynchronized DCC schemes (DccReactive-3 and -4) than for synchronized scheme (DccReactive-1 and -2). For timer handling, Cancel-and-Go schemes show poorer performances (DccReactive-2 in comparison to DccReactive-1 and DccReactive-4 in comparison to DccReactive-3).

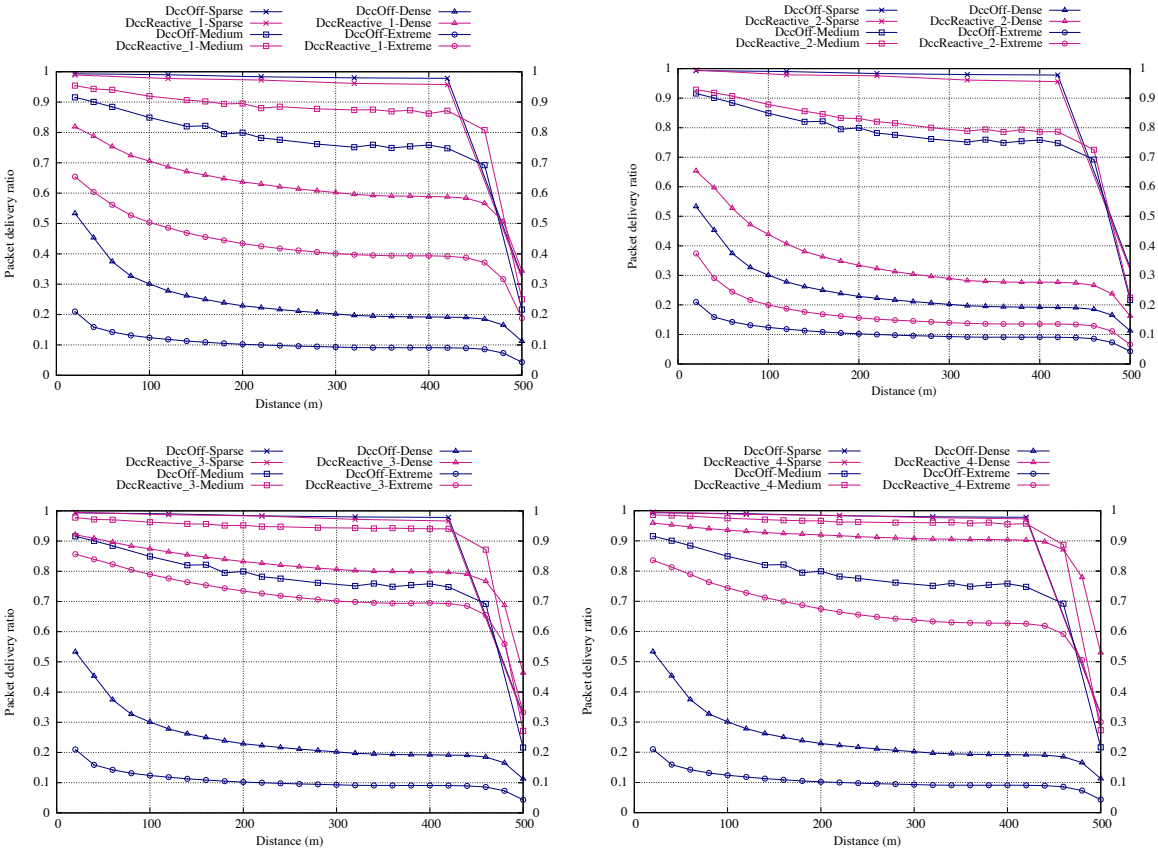

Figure 3 Comparison of Packet Delivery Ratio for different density classes.

Figure 4 plots the average packet inter-reception time (PIR) of the reactive DCC mechanisms in contrast to that of DccOff. Similar to the PDR case, DccOff shows an excellent PIR performance in the sparse scenario, but the performance largely degrades for higher density classes and it can exceed 1 second in the extreme density class. The reactive DCC mechanisms show better or worse PIR performances, depending especially on if the mechanism has synchronized or unsynchronized behaviour. Both the synchronized schemes, DccReactive-1 and DccReactive-2, in general, show poorer performance w.r.t DccOff, except that the case of DccReactive-1 (Wait-and-Go) in the extreme density class. On the other hand, the unsynchronized schemes, DccReactive-3 and DccReactive-4, provide improved performances for dense and extreme classes. The performance improvement is significant for the DccReactive-3 (Wait-and-go & Unsynchronized) and the performance degradation is significant for DccReactive-2 (Cancel-and-Go & Synchronized).

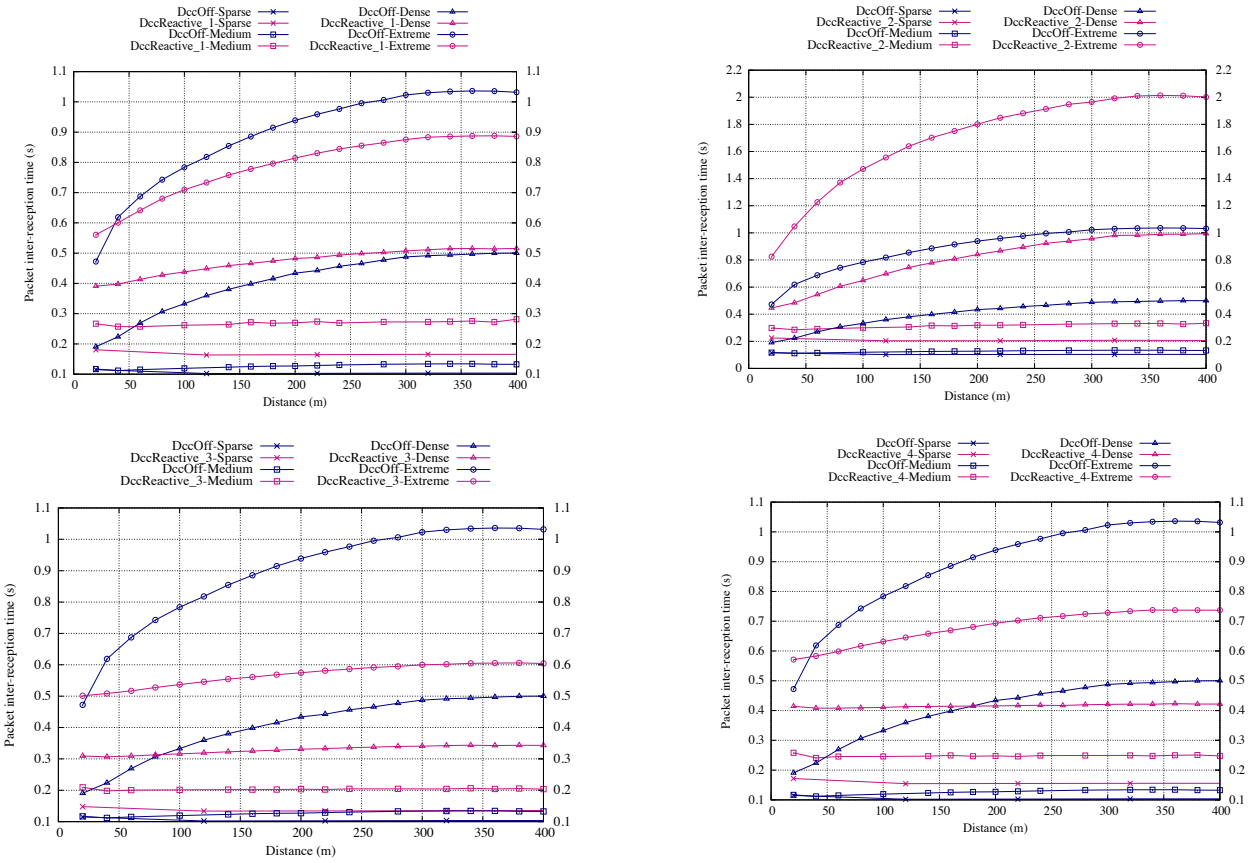

Figure 4 Comparison of PIR performances.

Figure 5 and Figure 6 plot the total number of transmissions and the channel load measured during a 5-seconds time interval for the dense scenario. Roughly speaking, the number transmissions during 20-milliseconds of time bins takes values in the range of [27, 35] for DccOff. In contrast, the value oscillates in the range of [5, 30], [0, 35], [10, 20] and [7, 12] for DccReactive-1, -2, -3, and -4, mechanisms respectively. Similar oscillated behaviours can be observed for the measured CBR (Figure 6). Specifically, in the dense scenario, CBR of DccOff is stable at 0.84%. In contrast, the CBR value oscillates in the range of [0.2, 0.8], [0.1, 0.7], [0.55, 0.8], and [0.4, 0.6] for DccReactive-1, -2, -3, and -4, respectively.

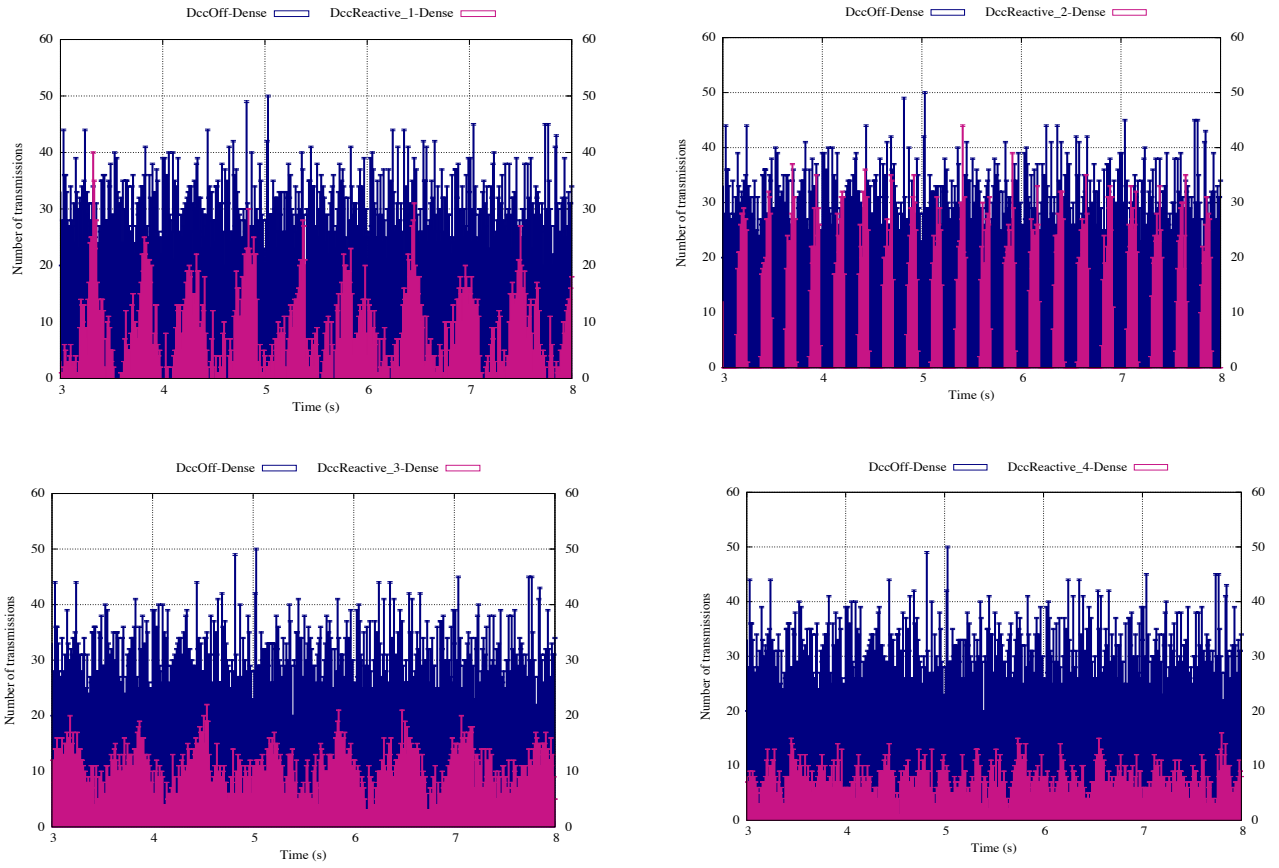

Figure 5 The distribution of the number of transmissions during a 5-second interval for dense scenario.

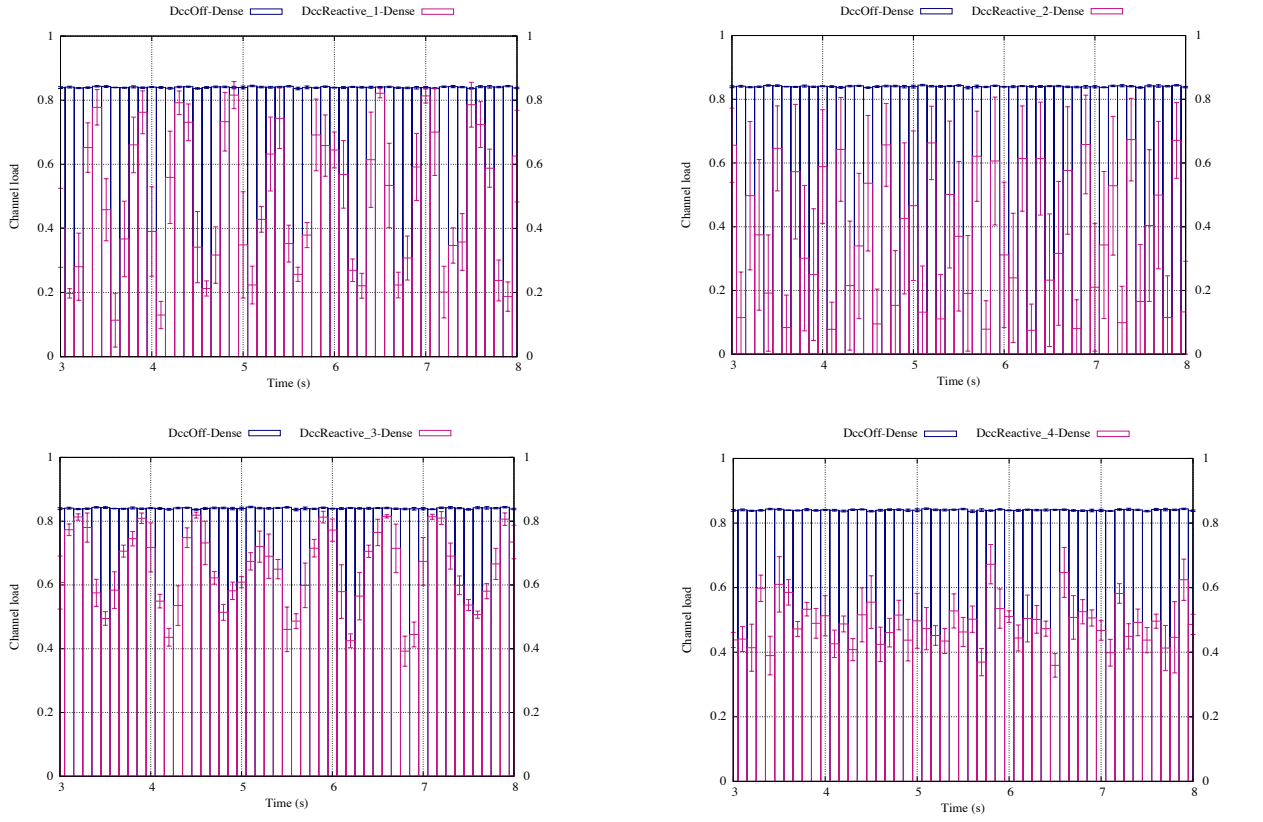

Figure 6 Average CBR during a 5-seconds of time interval.

Now we will have a closer look to the CAM generation behaviour at an ITS-S. Figure 7 plots the setting and actual values of the CAM intervals as well as the measured CBR at a randomly selected ITS-S in the dense scenario. (Note that for visibility reason, the parameters are plotted only when the values change.) Similar to what it is seen in Figure 6, CBR fluctuates more for the synchronized mechanisms and less for the unsynchronized mechanisms. The setting value of the CAM interval tend to jump between the highest (460 ms) and the lowest (60 ms) values of the parameter table, Table 2, for the synchronized mechanisms (DccReactive-1 and -2). In the unsynchronized mechanisms, the CAM interval was set to large values (above 260 ms). Finally, CAMs tend to be transmitted at the intervals 1) equal to the setting intervals for DccReactive-1, 2) longer than the setting interval for DccReactive-2, 3) shorter and the equal to the setting interval in DccReactive-3, and 4) shorter or larger than the setting interval in Dcc-Reactive-4. Longer intervvals than the setting values that observed in DccReactive-2 and -4 are conceivably due to the "Cancel-and-Go" behaviour. Shorter intervals than the setting values that observed in DccReactive-3 and -4 are due to the "unsynchronized" behaviour.

The obvious observation can be made from Figure 5 - Figure 7, is that the reactive DCC mechanisms in general use the wireless channel in an oscillated manner, and the oscillation is significant for the synchronized mechanisms (DccReactive-1 and DccReactive-2) especially for Cancel-and-Go & Synchronized scheme (DccReactive-2).

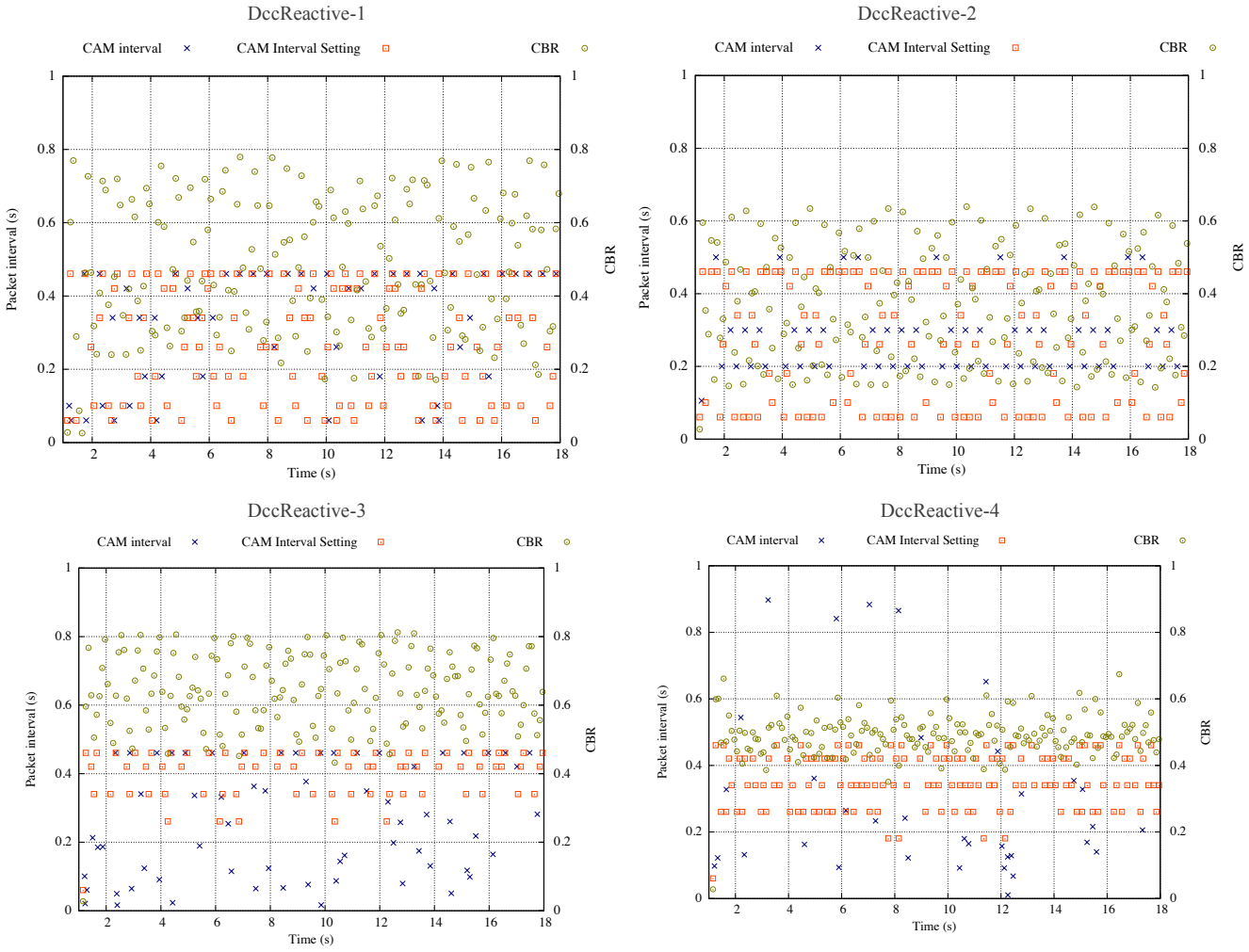

Figure 7 The setting (orange) and the actual (blues) values of the CAM interval and the measured CBR at a randomly selected ITS-S in the dense scenario.

4.3 Contribution 2: Study on channel load characterisation

In this subsection, we study the impact of the weight factor α of the channel load defined in (1). The study is made based on the performance investigations of DccReactive-1 and -3 for homogeneous static highway scenario. Note that we omit the Cancel-and-Go mechanisms (DccReactive-2 and -4), because our results presented in 1.5.2 show that Cancel-and-Go mechanisms show degraded performance compared to the Wait-and-Go schemes. For simplicity, we now call DccReactive-1 as *SyncDccReactive* and DccReactive-3 as *UnsyncDccReactive*.

Figure 8 and Figure 9 plot the PIR performances of SyncDccReactive and UnsyncDccReactive schemes for different density scenarios. Each graph has several curves for different transmitter and receiver distance ranges. Specifically, the curve for "d" indicates the results obtained for transmitters and receivers that are at the distance [d-20, d+20] m from each other. The horizontal axis is α (in percentage), and hence we are interested in α, which provides the smallest (or relatively small) PIR value. As can be seen in Figure 8, no particular value of α that provides satisfying performance can be found for SyncDccReactive for all the scenarios and distance ranges. We can say however, α =1, where the algorithm considers only the last CBR value, tends to lead to poorer performances for SynDccReactive. In contrast, for UnsyncDccReactive (see Figure 9), α =1 provides the best performance almost all the scenarios and distance ranges.

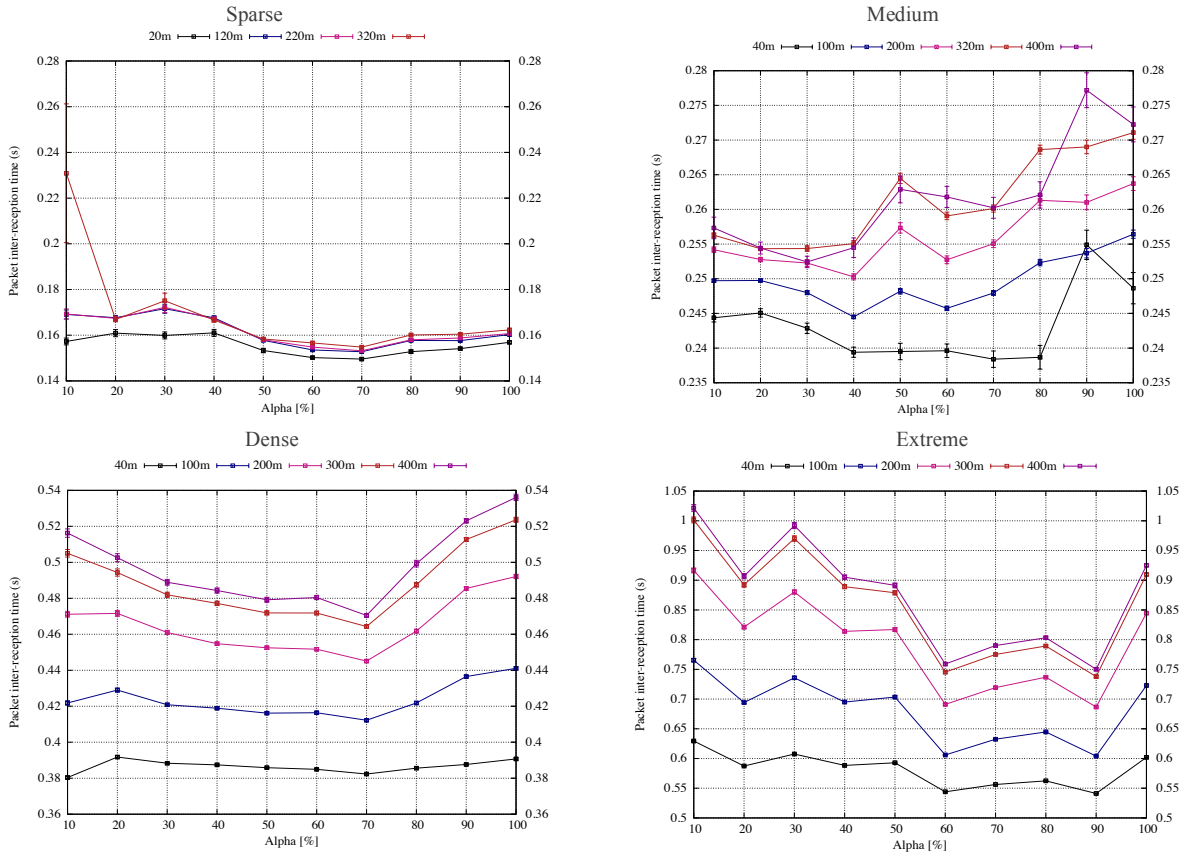

Figure 8 PIR of SyncDccReactive for different values of the weight factor (α).

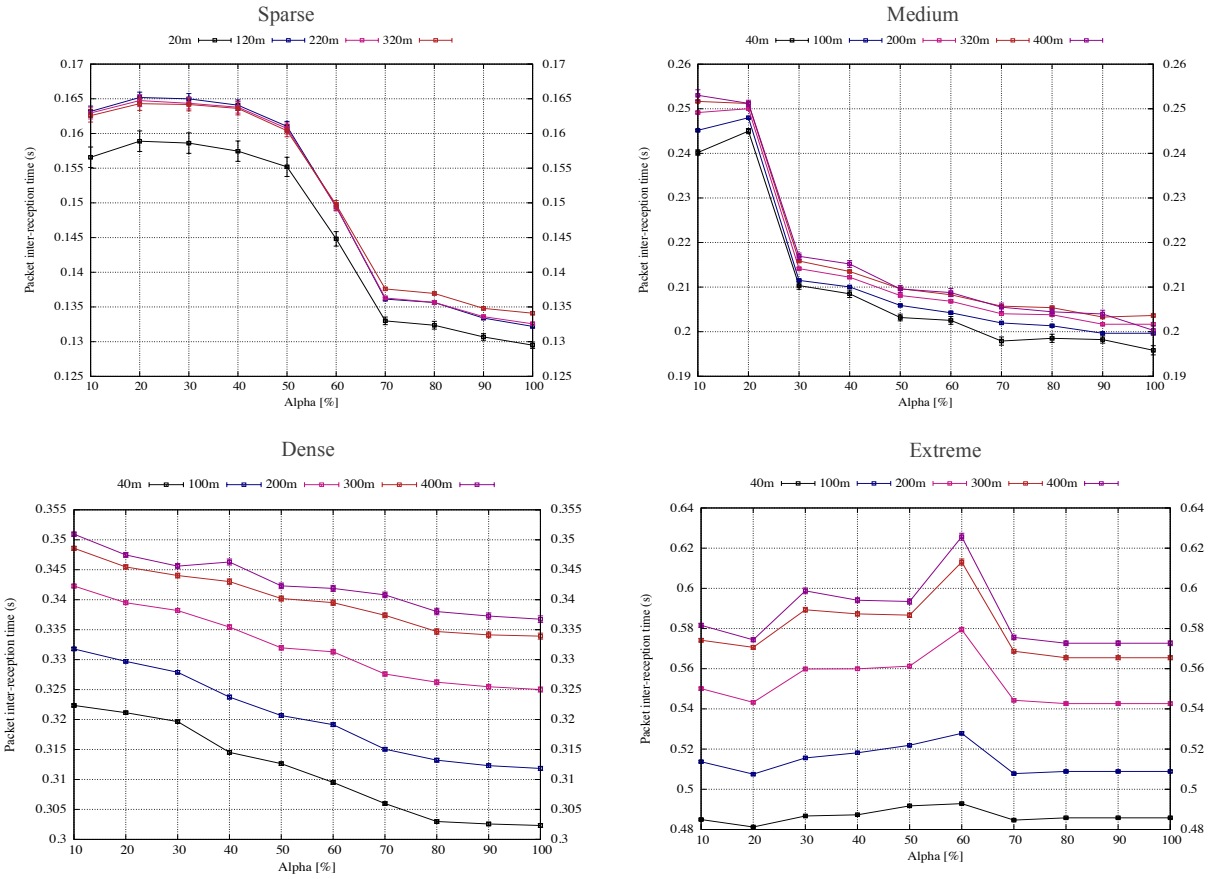

Figure 9 PIR of UnsyncDccReactive for different values of the weight factor (α).

4.4 Contribution 3: Study on non-identical sensing capabilities

In this subsection, we study the impact of non-identical sensing capabilities. Based on the previous work, this study targets DccReactive-3 (Wait-and-Go & Unsynchronized) and DccOff mechanisms.

Figure 11 and Figure 10 compare PDR and PIR of DccReactive and DccOff mechanisms for the cases where ITS-Ss have identical and non-identical sensing capabilities. The horizontal axis is the distance between the transmitters and the receivers. As can be seen in Figure 11, if the ITS-Ss have identical capabilities, the average PDR is stable when the distance between the transmitter and receiver is below 420 meters for both DccOff and DccReactive mechanisms. On the other hand, when the system consists of non-identical ITS-Ss, the stable distance is up to 250 meters for the sparse scenario and shorter for the medium and high density scenarios. A similar observation can be maid for PIR, which can reach 1.6 seconds for DccOff for non-identical sensing capability.

Jain's fairness index is calculated targeting the number of transmissions at individual mobile ITS-S. The horizontal axis shows the density classes: 100m, 45m, 20m, and 10m represent the sparse, medium, dense, and extreme classes. In the case of the identical ITS-Ss, the fairness index is 100% for DccOff regardless of density class, and it is slightly lower for DccReactive. In contrast, in the case of non-identical ITS-Ss, the fairness index degrades, and performance degradation is significant for DccReactive.

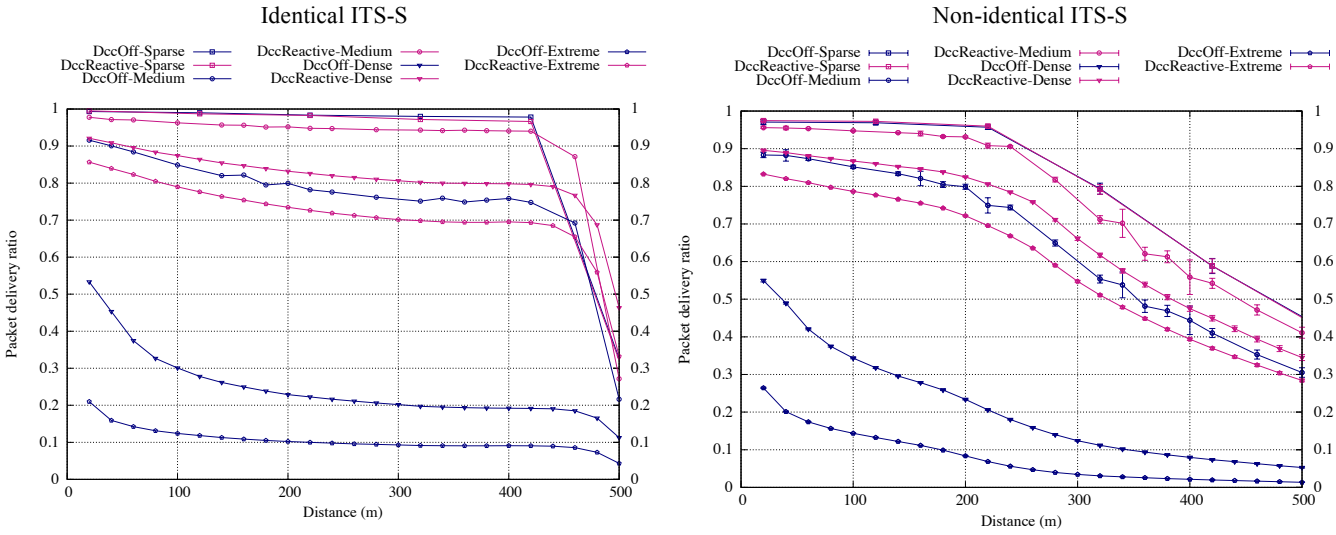

Figure 11 Comparison of PDR for the systems with ITS-Ss with identical and non-identical receiver capabilities.

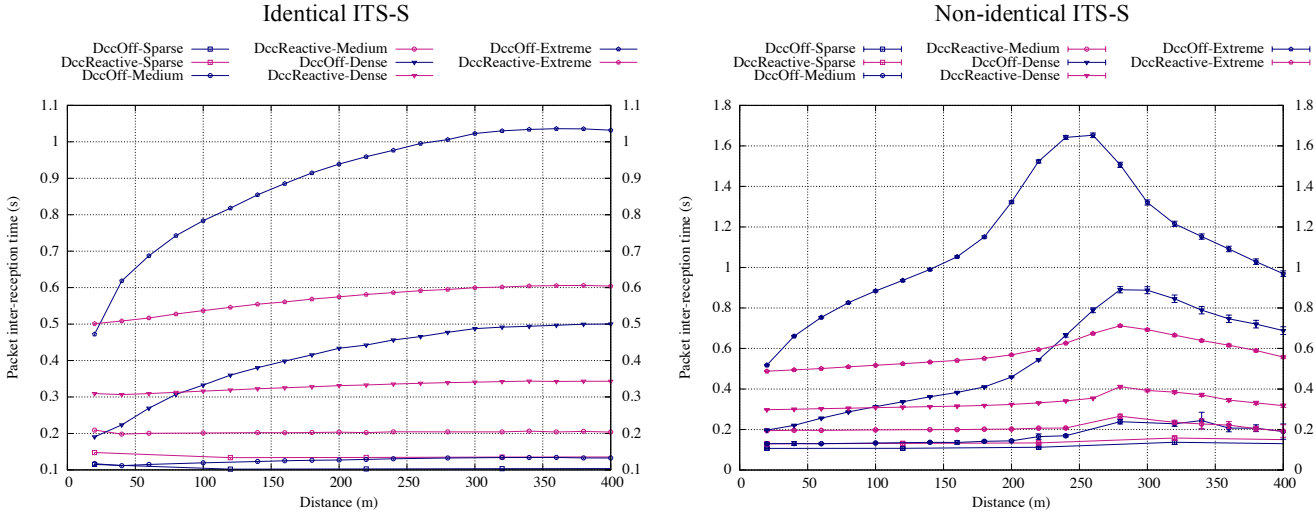

Figure 10 Comparison of PIR for the systems with ITS-Ss with identical and non-identical receiver capabilities.

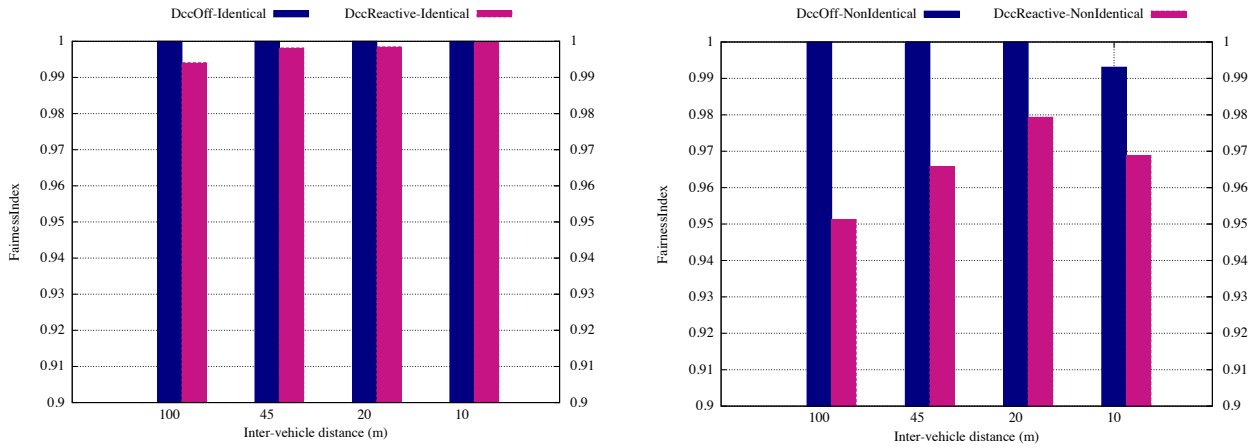

Figure 12 Comparison of fairness index for the systems with ITS-Ss with identical and non-identical receiver capabilities.

5 Conclusions

In what follows, we conclude our studies on 1) DCC synchronization 2) channel load characterisation, and 3) non-identical sensing capability issues.

5.1 Contribution 1: Study on synchronization issue of DCC

The table below lists the maximum positive and negative performance differences between the individual DccReactive and DccOff for the homogenous highway scenario. The table shows that in terms of PDR, DccReactive-4 (Cancel-and-Go & Unsynchronized) shows the best performance; but in terms of PIR the DccReactive-3 (Wait-and-Go & Unsynchronized) outperforms the other mechanisms. Since PDR is the ratio of the number of the received and the transmitted packets, a large PDR can be obtained by aggressively reducing the number of transmissions, i.e., CAM message generations. This is not always a good solution for safety applications. Indeed Figure 5 clearly shows that the number of transmissions of DccReactive-4 is significantly smaller than that of DccReactive-3, explaining why DccReactive-4's PDR is better than that of DccReactive-3. On the other hand, PIR is the length of time during which the receiver node does not receive data from the transmitter node. For CAM packets, this obviously implies the time gap during which the receiver ITS-S does not have information about the transmitter ITS-S. Therefore PIR is one of the key parameters that can determine whether the V2X communications protocol can support such safety applications. To this reason, by paying more attention on PIR performances, and we conclude that DccReactive-3, Wait-and-Go & Unsynchronized, mechanism is the best approach among the four versions of DccReactive.

Table 5 Maximum performance improvement/deterioration of DccReactive schemes w.r.t DccOff.

Algorithms	PDR difference ($PDR_{DccReactive} - PDR_{DccOff}$)		PIR difference ($PIR_{DccReactive} - PIR_{DccOff}$)	
	Max improvement (Positive difference) [%]	Max deterioration (Negative difference) [%]	Max improvement [s]	Max deterioration [s]
DccReactive-1	44	-2	0.22	-0.43
DccReactive-2	16	-2	0	-1.16
DccReactive-3	68	-1	0.68	0.22
DccReactive-4	71	-0.5	44	0.36

5.2 Contribution 2: Study on channel load characterization

Table 6 and Table 7 lists the weight factor, α (see (1)), which corresponds to the shortest PIR for SyncDccReactive (DccReactive_1) and UnsyncDccReactive (DccReactive_3) mechanisms. The difference of PIR between the minimum value and that for α=1 is indicated for the cases, where α is not 1. Table 6 clearly shows that for the synchronized DCC system, the best PIR performances never achieved when α=1; the difference between the minimum PIR and that for α=1 is over 10 milliseconds. This implies that it is difficult to characterize channel load by only the current CBR. In contrast, as Table 7 shows that when the system is unsynchronized, the best PIR is achieved when α=1, indicating that channel load can be characterized by only the current CBR if the system is unsynchronized. Note that α=0.2 provides the shortest PIR for small transmitter and receiver distances in the extreme scenario for UnsyncDccReactive, the PIR difference between the minimum value that that for α=1 is very small (4ms).

Table 6 The weight factor, α, which corresponds to the shortest PIR for SyncDccReactive.

Scenario	Transmitter and Receiver distance range [m]				
	40 m	100	200	300	400
Medium	0.7 (10ms shorter than $PIR(\alpha = 1)$)	0.4 (12ms shorter than $PIR(\alpha = 1)$)	0.4 (13ms shorter than $PIR(\alpha = 1)$)	0.2 (16ms shorter than $PIR(\alpha = 1)$)	0.3 (20ms shorter than $PIR(\alpha = 1)$)
Dense	0.2 (10ms shorter than $PIR(\alpha = 1)$)	0.1 (29ms shorter than $PIR(\alpha = 1)$)	0.7 (47ms shorter than $PIR(\alpha = 1)$)	0.7 (59ms shorter than $PIR(\alpha = 1)$)	0.7 (66ms shorter than $PIR(\alpha = 1)$)
Extreme	0.1 (61ms shorter than $PIR(\alpha = 1)$)	0.9 (12ms shorter than $PIR(\alpha = 1)$)	0.9 (16ms shorter than $PIR(\alpha = 1)$)	0.9 (17ms shorter than $PIR(\alpha = 1)$)	0.9 (17ms shorter than $PIR(\alpha = 1)$)

Table 7 The weight factor, α, which corresponds to the shortest PIR for UnsyncDccReactive

Scenario	Transmitter and Receiver distance range [m]				
	40	100	200	300	400
Medium	1	1	1	1	1
Dense	1	1	1	1	1
Extreme	0.2 (4 ms shorter than that of $\alpha = 1$)	0.2 (4 ms shorter than that of $\alpha = 1$)	1	1	1

5.3 Contribution 3: Study on non-identical sensing capability

Figure 11 and Figure 10 clearly show that when the ITS-Ss have non-identical sensing capabilities, in general, the average communication range tends to be reduced for both DccOff and DccReactive schemes. The unfair sensing capabilities result in unfair transmission behaviours. The unfairness issue is more significant for DccReactive than for DccOff.

6 Summary

In this work, we studied the following issues targeting reactive dynamic DCC algorithm.

- Synchronization
- Channel load characterization
- Non-identical receiver capability

Following conclusions were drawn:

- It is very important to provide a solution to avoid synchronized DCC behaviour among ITS-Ss. If a careful attention is given on this issue, the simple reactive DCC algorithm can perform better than DccOff. In the case of rate adaptation, introducing a random rate seems to be a good solution.
- If the road traffic is sparse, the reactive DCC algorithm tends to show poorer performance than DccOff.
- Cancelling timer for the CAM generator seems to be not necessary.
- If the system is unsynchronized, it seems that only the current CBR can be a good indicator of the channel load. However, if the system is synchronized, it is necessary to pay attention on CBR for longer interval.
- If the system consists of ITS-Ss with heterogeneous channel sensing capability, non-negligible negative impact can be expected in terms of communications range and fairness.
- The fairness issue caused by non-identical sensing capabilities is more significant for DCC-enabled system.

References

[1] ETSI TS 102 687 V1.1.1 ITS; Decentralized Congestion Control Mechanisms for Intelligent Transport Systems operating in the 5 GHz range; Access layer part, 2011.

[2] Kenney. J.B, Bansal. G, Rohrs. C.E, LIMERIC: a linear message rate control algorithm for vehicular DSRC systems, In 8th ACM Int. Workshop on Vehicular Inter-networking VANET'11, pp. 21-30, 2011.

[3] Ruehrup. S, Fuxjaeer. P, Smely. D, TCP-Like Congestion Control for Braodcast Channel Access in VANETS, in 3rd Int. Conf. on Cennected Vehicles and Expo (ICCVE) 2014.

[4] NS3, network simulator, http://www.nsnam.org

[5] SUMO, Simulation of Urban mobility, http://www.dlr.de/ts/en/desktopdefault.aspx/tabid-9883/16931_read-41000/

[6] NS3, WAVE module, http://www.nsnam.org/docs/models/html/wave.html