Open Archive TOULOUSE Archive Ouverte (OATAO)

OATAO is an open access repository that collects the work of Toulouse researchers and makes it freely available over the web where possible.

Any correspondance concerning this service should be sent to the repository administrator: staff-oatao@listes-diff.inp-toulouse.fr

Improving web experience on DVB-RCS2 links

Nicolas Kuhn[1,2,3] Olivier Mehani[3] Huyen-Chi Bui[2] Emmanuel Lochin[2]
Jérôme Lacan[2] José Radzik[2] Roksana Boreli[3]

[1] IMT Telecom Bretagne, IRISA, France
[2] Université de Toulouse, ISAE-SUPAERO, France
[3] National ICT Australia, Australia

Abstract

The specifications of Digital Video Broadcasting - Return Channel via Satellite (DVB-RCS2) state that the satellite gateway could introduce both random and dedicated access methods to distribute the capacity among the different home users. Before starting an engineering process to design an algorithm allowing to combine both methods, it seems necessary to assess the performance of each. This paper compares random and dedicated access methods by measuring their impact on the performance of Transmission Control Protocol (TCP) sessions when the home users exploit the DVB-RCS2 link for regular use (*e.g.*, web browsing or email transmission).

In this paper we detail the implementation of an NS-2 module emulating Physical Channel Access (PCA). This module fills a gap in terms of random and deterministic access methods and allows to model various satellite channel access strategies. Based on NS-2 simulations using realistic system parameters of the DVB-RCS2 link, we demonstrate that, compared to dedicated access methods, which generally result in higher levels of transmitted data, random access methods enable faster transmission for short flows. We propose to combine random and dedicated access methods, with the selection of a specific method dependent on the dynamic load of the network and the sequence number of the TCP segments.

1 Introduction

The Digital Video Broadcasting Project (DVB) is a consortium committed to designing standards for video and data service. The satellite service of DVB is divided between the transmission system (from providers to users) and the control channel (from users to providers). The current standards for the satellite links are DVB-S2 (diffusion channel) and DVB-RCS (return channel). In 2013, DVB validated the specifications for the second generation of the satellite return channel, DVB-RCS2. In [2], an overview of the system is given, [3] details the standards for satellite lower layers, and [1] details higher layers specifications. This new version for the transmission of data from home users to satellite gateways features enhanced security, improved Quality of Service and support for IPv6. This would enable the user terminals to not only transmit control data, but also send HTTP requests and other low volume uplink Internet traffic.

The quality of experience for a user browsing the web or sending an email is highly linked to the latency. As an example, in [27], the authors explain that Google measured that "an additional 500 ms to compute (a search) [...] resulted in a 25% drop in the number of searches done by users". To improve the web experience over DVB-RCS2 links, we propose to adapt the channel access strategy and reduce the transmission time of the first packets of a transmission. While the standard argues that both methods (switching between random and dedicated) can be implemented, no insight is provided as to their potential benefits. As a matter of fact, before starting an engineering process to design an algorithm allowing to combine both methods, it seems necessary to assess the performance of each.

Preliminary studies [23, 22] have presented the simulating tool PCA and have evaluated the use of random access methods in the context of DVB-RCS2 system. In this paper, we extend these works

by providing an in-depth analysis of the utility of driving the selection of DVB-RCS2 channel access method by using transport layer (TCP) protocol information.

The remainder of the article is organized as follows. We present the different component of the DVB-RCS2 specifications in Section 2. We also summarize the main studies assessing the impact of access methods on the performance of transport layer protocols. In Section 3, we present our NS-2 module that enables modelling of the DVB-RCS2 channel access, by leveraging experimental data. Section 4 presents the network parameters used in this study. To compare the dedicated and random access methods, in Section 5 we evaluate the overall network performance, for a scenario that includes multiple TCP sessions. We provide further insights into TCP performance with different access methods in Section 6, where we evaluate the time needed to transmit the first packets of a TCP session. We discuss additional issues related to switching between different access methods in Section 7 and conclude in Section 8.

2 State of the art

2.1 Multi-Frequency Time Division Multiple Access (MF-TDMA)

On an MF-TDMA link, the capacity is shared at the *Access Point*. The access point forwards traffic between one or more satellite gateway and satellite terminals (home users) over the shared medium, therefore covering both up and down link scenarios. We provide some definitions of the terms used in this paper to describe MF-TDMA processes:

- Flow: data transfer at the transport layer;

- Datagram: network layer segment of a flow;

- Link Layer Data Unit (LLDU): N_{data} bytes of a fragmented datagram;

- Physical Layer Data Unit (PLDU): LLDU with an optional N_{repair} recovery bytes ($N = N_{\text{data}} + N_{\text{repair}}$);

- Block: PLDUs can be split into N_{block} blocks if the access method requires;

- Frame: "time $\times$ frequency" set of blocks transmitted between gateway and users, generated every T_F;

- Slots: element of a frame where a block can be scheduled.

satellite gateway and the home users (Satellite Terminals, ST): the satellite medium is shared among the users and protocols must allocate resources to each of them. The channel access methods define the way data can be transmitted from the home users (Satellite Terminals, ST) to the satellite gateway, exploiting the DVB-RCS2 link. From the satellite gateway to the ST, Time-Division Multiplexing is used, guaranteeing a sufficiently high capacity for each terminal. Te return link (DVB-RCS2) is therefore the performance bottleneck of the connection in terms of delay.

The resource is distributed every $T_F = 45\,\text{ms}$ by the Network Control Center (NCC). The NCC is the element that adapts the repartition of the available slots at each frame, in order to (1) accept late-comer flows; (2) adapt the time slots reservation depending on each user characteristics (different priority between the users); (3) adjust the distribution of time slots depending on the network load; (4) optimize the "mod-cod" (modulation and coding) at the physical layer for dedicated access methods [2]. The NCC transmits a "Burst Time Plan"(BTP) to the users and indicates when and how to transmit data. Several timeslots are available per frequency. A "time $\times$ frequency" block is called a "frame" [3, sec. 7.5.1.1], detailed in Figure 1.

Every time slot, an access scheme may allow terminals to transmit data on a given sub-"time $\times$ frequency" block. There are more blocks than carriers and we refer to these blocks simply as "frequencies" in the rest of the document. We denote by N_S the number of time slots available per frequency. The frequencies on which data is transmitted can be divided depending on the access method: F_R frequencies are dedicated to the random access methods and F_D are reserved to the
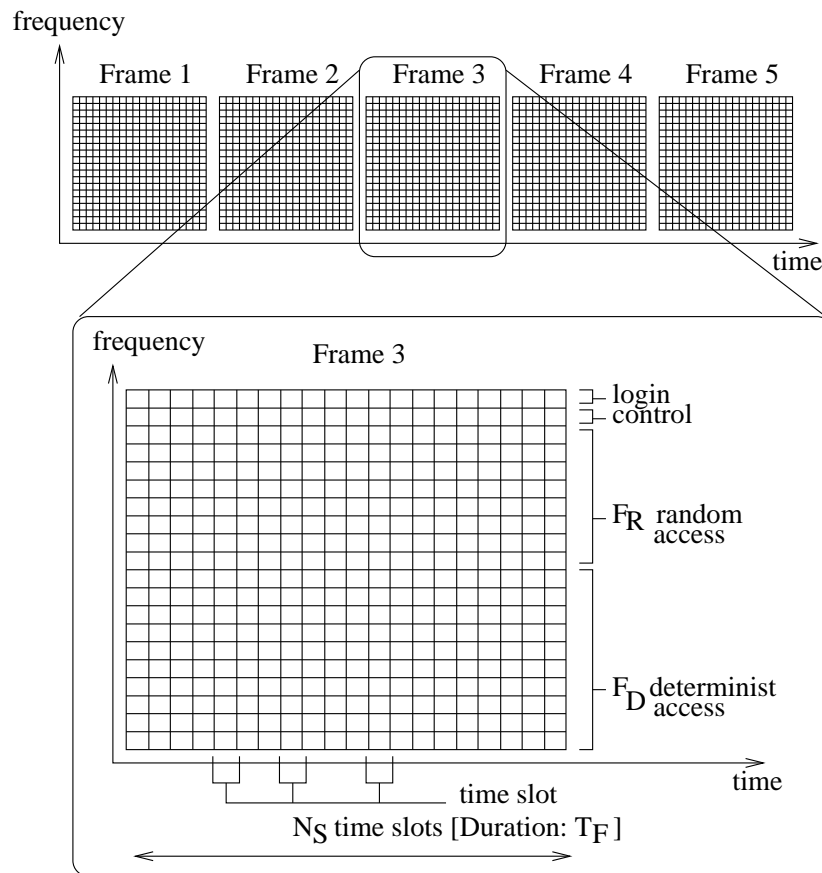
Figure 1: Times $\times$ Frequency block of time slots

dedicated access methods. In total, a frame can carry $N_S \times (F_R + F_D)$ slots. The BTP specifies which timeslots each user can transmit data on [3, sec. 6.2.2.8]. The resource distribution depends on the nature of the flows, the network load and the access methods. The standard defines two strategies for channel access:

- Dedicated Access [3, sec. 7.2.6]: timeslots are reserved for the transmission between ST and terminals. This induces a negotiation delay.

- Random Access [3, sec. 7.2.5]: in order to simplify and minimize the negotiation at the MAC level, there is no reservation of timeslots.

2.1.1 Capacity allocation schemes

DVB-RCS2 features capacity allocation schemes that control how the capacity is given to an ST: Continuous Rate Assignment (CRA), Rate-Based Dynamic Capacity (RBDC), Volume-Based Dynamic Capacity (VBDC) and Free Capacity Allocation (FCA). CRA, RBDC and VBDC are dedicated access methods, as they are the result of specific user requests, whereas FCA is a random access method. The rationale of this paper is to assess how TCP congestion and flow control are affected by dedicated and random access methods. For the sake of simplicity, we consider the CRA and FCA capacity allocation schemes, as they are basic allocation schemes that would not increase the complexity the simulation outputs, as RBDC and VBDC would. It is worth pointing out that RBDC and VBDC could easily be integrated in our NS-2 module presented in Section 3 and [23]. We note that considering RBDC and VBDC would result in slightly different numbers, but considering only CRA and FCA nonetheless allows us to derive qualitative conclusions on the respective impact of dedicated vs. random access method on TCP.

2.1.2 Dedicated Access with CRA

Depending on the load on the network, the NCC computes an adequate BTP for each ST having requested satellite capacity and established the connection [3, sec. 7.2.6.3]. Therefore, these methods ensure a reliable transmission of data but introduce a negotiation delay of at least one RTT. The reservation ensures that capacity is fairly distributed: if there are 40 slots available and 10 users, each user can transmit data on 4 slots.

2.1.3 Random Access with FCA

With Random Access methods, in order to simplify and minimize the negotiation at the MAC level, there is no reservation of timeslots. The NCC can not ensure that different terminals transmit data on separate blocks, which cannot guarantee a reliable transmission. Stronger error codes are therefore introduced at the physical layer: N_{repair} redundancy bytes are added to the reduced N_{data} bytes to form a code word of $N = N_{\mathrm{data}} + N_{\mathrm{repair}}$ bytes that is split into N_{block} blocks. N_{ra} slots form a *Random Access* block (RA block) on which erasure codes are introduced [3, sec. 7.2.5.2]. Each transmitter randomly spreads its N_{block} blocks across the N_{ra} slots of the RA block for spectral diversity. In [20], the authors define guidelines to design Random Access methods, and assess the performance of CRDSA [12]. Among the different Random Access methods, we can cite the following: Multi-Slots Coded ALOHA (MuSCA) [10], ALOHA [4], Diversity Slotted ALOHA [16], Contention Resolution Diversity Slotted ALOHA (CRDSA) [12]. Random access methods are not only exploited in the context of DVB-RCS2, but also in vehicular networks [8] and sensor networks [18].

Performance of random access methods can be described by the probability that a receiver decodes its N_{data} useful bytes depending on the number of users that transmit data on a RA block.

In [3, sec. 7.2.5.1.3] the authors advise that "the ST shall by default not transmit in contention timeslots for traffic, but may do this when explicitly allowed by indication in the Lower Layer Service Descriptor or by other administrative means" making the random access methods used mostly for login procedure [3, sec. 9.2.3] and optionally for traffic. Moreover, in [3, sec. 6.2.2.8], the specifications defines that the "Terminal Burst Time Plan Table version 2" (TBTP2) "may be used to assign dedicated access timeslots, [...] allocate timeslots for random access" and indicate the access methods used for each

timeslots information to the BTP. The specifications present the potential for introducing both random and dedicated access methods without detailing in which proportion the timeslots of the frames must be divided between them. even though [3, sec. 7.2.5.1.3] highlights the preference for dedicated access methods.

2.2 Existing studies on the impact of DVB-RCS2 channel access methods on TCP

2.2.1 TCP and Dedicated Access

In [28], the authors analyze the interactions between the TCP and DVB-RCS control loops by exploring the performance of TCP over the different capacity allocation categories defined in the DVB-RCS standard. They show that the performance of access schemes is strongly linked to the traffic characteristics such as the size of the flow, or the required QoS. In [26], the authors highlight that, in the context of Demand Assigned Multiple Access (DAMA), delay variability severely impacts the performance of TCP. They run simulations with NS-2 and analyze the MAC–TCP interactions to improve the performance of TCP New Reno. Then they propose a cross-layer technique based on queuing sizes at the MAC layer. Their proposal is not adapted to short flows nor to the DVB context and focuses only on dedicated access methods. In [6], the author analyses the performance of competing TCP flows using different return channel satellite terminals (RCSTs) and competing for the DVB-RCS return link through DAMA mechanisms. They consider an emulated network and observe relatively poor performance. They only focus on the dedicated access method.

The studies presented above [28, 26, 6] focus on dedicated access methods and highlight the difficulties encountered by TCP to transmit data on the return link. They show the negative impact of the queuing delay introduced by access methods, without comparing their results when random access methods are applied.

2.2.2 TCP and Random Access

In a mobile context (mobile cars and satellite links), the authors of [14], show that, when users act as senders, random access methods are not suitable; however, depending on the size of the file transmitted, there is a certain interest for dedicating more timeslots for random access methods when the users act as receivers. Their results can not be leveraged in our specific context because of the model for satellite mobile links and the different capacities and access strategies do not apply. However, following this idea, the authors of [30] highlight a possible advantage of introducing more random access methods in DVB-RCS2. More recently, the authors of [13] assess the issues encountered by TCP over CRDSA in the context of DVB-RCS2. However, they do not conduct extensive simulations neither in terms of traffic considered nor channel access strategies, which make their studies insufficient to properly determine the benefits of using random schemes instead of dedicated schemes.

In [5], the authors evaluate how the loss events with random access methods impact on TCP performances when TCP NewReno is used to carry sustained rate traffic. They conclude that there is an interest in using random access schemes for such small flows, which is an interesting input for cross-validating our results. However, this work is not entirely in the scope of our paper, as: (1) we intend to evaluate the impact of various access methods (dedicated and random) on the internal parameters of TCP, such as its congestion window evolution; (2) the metrics that they provide hardly let us assess the latency for short flows.

The analysis of the bad performance of TCP on high variable delays is relevant [28] and reducing the connection establishment delay with random access methods might be of interests [30, 13, 5]. However, existing studies do not enable to provide a definitive answer as to whether introducing random access methods to carry data traffic is beneficial. As far as we know, there is a lack of relevant comparisons of the various channel access strategies in the context of DVB-RCS2.

2.3 Simulating DVB-RCS2

We now survey available tools to simulate DVB-RCS2.

As discussed in [25], emulation can be expensive not only for the technology involved but also for the "man-in-the-loop" manipulations and synchronization. Additionally, new satellite transmission may require not-yet-validated or -implemented support. In addition to this we assess experimental access methods and, therefore prefer running parametrable simulations to experimenting on less flexible real deployments of DVB-RCS2 networks. A few models of the DVB-S2/RCS satellite network in NS-2 have already been proposed in [19, 29]. These NS-2 modules attempt to be as close as possible to the real system. While accurate, they are however not flexible enough for our study, as we want to integrate specific inputs, such as performance of experimental random access methods, other internal parameters of the NCC component. For this purpose, we implemented a specific module to emulate channel access on top of experimental Physical Channel Access (PCA) traces.

3 Physical Channel Access (PCA)

In Figure 2, we compare the `enque()` and `deque()` methods of `DropTail` and `DropTail/PCA`. With `DropTail`, when the `enque()` method adds packet P_{N+1}, it is added at the end of the sending buffer and transmitted when $P_1, \ldots, P_N$ have been transmitted with the `deque()` method. With `DropTail/PCA`, when a packet is `enque()`ed, it is also added to the sending buffer. However, depending on the access method introduced, only a subset of the datagram is considered sent with each frame. Each subset of a datagram will be transmitted with the same access method, and when the last byte of a datagram has been transmitted, `deque()`, which is called every T_F, removes the packets from the sending buffer and passes it along. More details on the implementation of PCA can be found in [23].

4 Simulation parameters

4.1 Framing parameters

The following parameters have to be specified prior to starting a simulation:

- `cutConnect_`: time after which the connection between the gateway and the user is closed (in seconds);

- `esN0_`: signal-to-noise ratio of the channel in dB;

- `switchAleaDet_`: sequence number at which the access method switches from random to dedicated (see Section 7);

- `frameDuration_`: duration of a frame (T_F);

- `nbSlotPerFreq_`: number of time slots per frequency (N_S);

- `sizeSlotRandom_`: useful number of bits that can be sent on one RA block (*i.e.*, where random access methods are introduced) (N_{data});

- `sizeSlotDeter_`: useful number of bits for each time slots where dedicated access methods are introduced (N_{data});

- `rtt_`: two-way *link* delay (in seconds);

- `freqRandom_`: number of frequencies used for random access (F_R);

- `nbFreqPerRand_`: number of frequencies comprised in an RA block (($F_R \times N_S$)/N_{ra});

- `freqDeter_`: number of frequencies used for dedicated access (F_D);

- `maxThroughtput_`: maximum authorized throughput for one given flow (in Mbps);

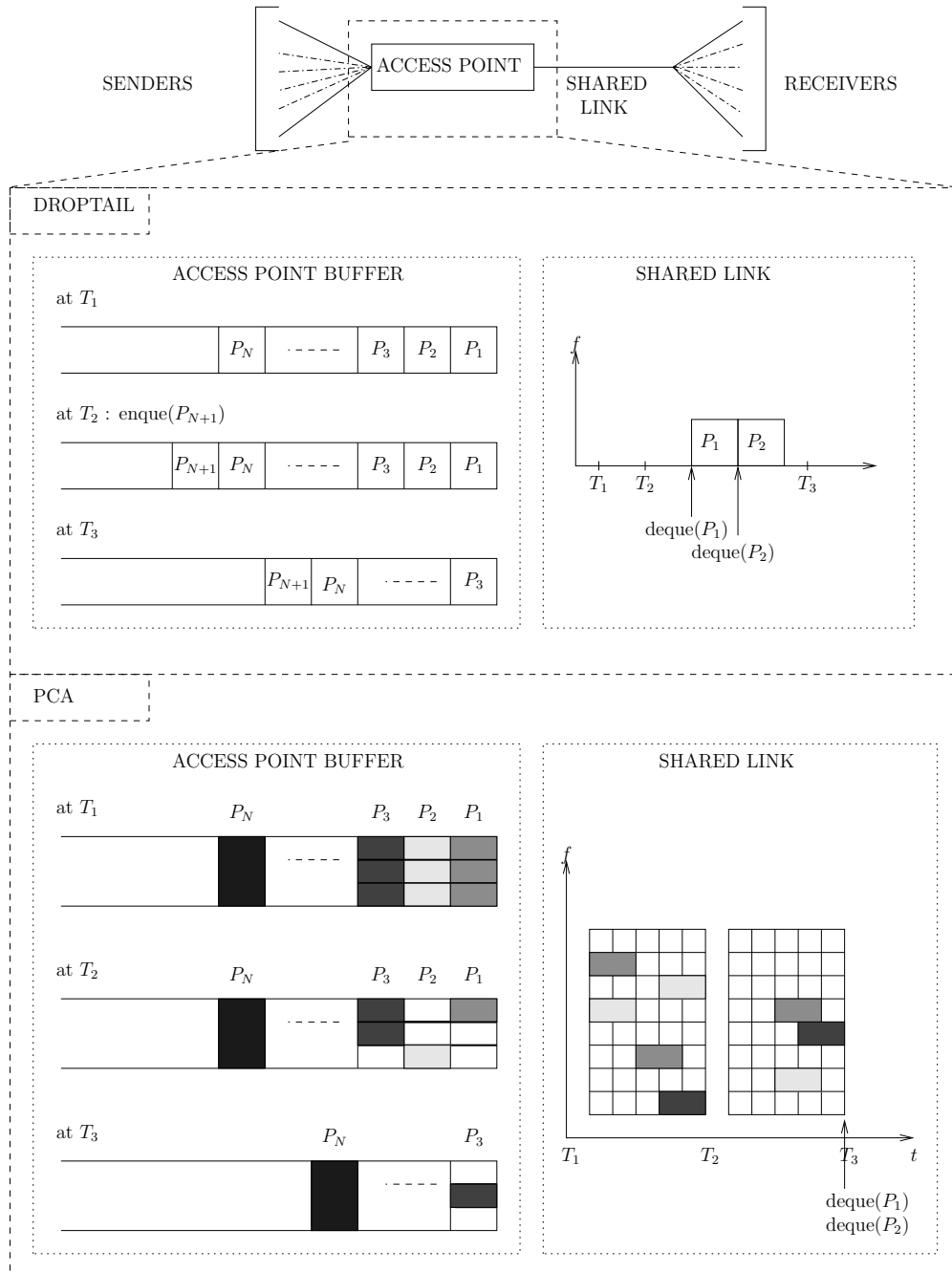- `nbSlotRndFreqGroup_`: number of blocks a PLDU is split into for distribution in one RA block (N_{block});

Figure 2: `enque()` and `deque()` methods: capacity allocation with DropTail and PCA

- `boolAntennaLimit_`: boolean indicating whether one transmitter has one (False) or $F_R + F_D$ (True) single terminal amplifier (see Section 4.2).

4.2 Single terminal amplifier limitation

In DVB-RCS2, the outdoor unit includes a unique RF power amplifier: therefore, terminals cannot send data on different frequencies at once. This limitation imposes a maximum number of slots that a user is allowed to occupy on each frame. Also, the standards limits the hopping distance while switching frequencies: this is neglected in our model.

As an example, if $N_S = 40$ slots and:

- with a dedicated access, a unique user can use $N_S = 40$ slots;

- with a random access, $N_{block} = 3$, $N_{ra} = 40$: a unique user can use $\lfloor N_S/N_{block} \rfloor = 13$ slots.

4.3 Parameters

We consider a frame of $45\,\text{ms}$ length which covers 100 frequencies (*i.e.* comprising 100×40 blocks). Therefore, 100 slots are grouped in a random access (RA) block composed of 2.5 frequencies. We base the choice of parameters on specifications defined in [3] and present them in Table 1. We provide more details about `sizeSlotRandom_` and `sizeSlotDeter_` in Section 4.4.

Table 1: Use case simulation parameters

Parameters	Access method		
	Dedicated	**Random** CRDSA	**Random** MUSCA
`cutConnect_`	3	3	3
`esNO_`	5	5	5
`switchAleaDet_`	0	∞	∞
`frameDuration_`	0.045	0.045	0.045
`nbSlotPerFreq_` (N_S)	40	40	40
`sizeSlotRandom_` (N_{data})	xx	597	594
`sizeSlotDeter_` (N_{data})	920	xx	xx
`rtt_`	0.5	0.5	0.5
`freqRandom_` (F_R)	0	100	100
`nbFreqPerRand_`	2.5	2.5	2.5
`freqDeter_` (F_D)	100	0	0
`maxThroughtput_`	$1\,\text{Mbps}$	$1\,\text{Mbps}$	$1\,\text{Mbps}$
`nbSlotRndFreqGroup_` (N_{block})	1	3	3
`boolAntennaLimit_`	1	1	1

4.4 Access methods

4.4.1 Dedicated access

Each slot of length $1.09\,\text{ms}$ carries 536 symbols. In the simulations, we consider a "clear sky" scenario with a Signal-to-Noise-Ratio equal to $8.6\,\text{dB}$. We assume that the users apply a code of rate $R = 2/3$ combined with 8PSK modulation to encode a packet of 920 information bits into a codeword of 1,380 bits, *i.e.*, 460 symbols. Due to the encapsulation at the physical layer level, the physical layer data unit is increased to 536 symbols.

4.4.2 Random access

Users connecting to the satellite with random access generally use an operating point lower than for the dedicated access. In the rest of this article, we take a margin of 3.5 dB. Thus, for the "clear sky" scenario, we consider that: $(E_s/N_0)_{random} = (E_s/N_0)_{dedicated} - 3.5 = 8.5 - 3.5 = 5\,\text{dB}$. In systems using CRDSA at 5 dB each user can apply error-correcting codes of rate $R_{CRDSA} = 2/3$, associated with QPSK to encode a packet of 613 bits (597 information bits and 16 header bits) into a codeword of 920 bits, $i.e.$ 460 symbols. The error-correcting code used is a turbo code. As detailed in [12], N_{block} bursts of length about 530 symbols are then created. The number of generated bursts depends on the version of CRDSA. In this paper, we study the performance of regular CRDSA-3 ($N_{block} = 3$). The N_{block} bursts are transmitted randomly into N_{block} slots of an RA block.

With MuSCA [9, 10] as the random access method, users encode a packet of 680 bits (594 information bits and 86 header bits) with a turbo code of rate 1/4 associated with QPSK modulation to create codewords of 1,380 symbols. The codeword is split into $N_{block} = 3$ parts to generate $N_{block} = 3$ bursts sent on time slots of the same RA block.

Figure 3 depicts the performance in terms of packet loss ratio (PLR) depending on the number of packets transmitted per RA block by CRDSA-3 and MuSCA-3.
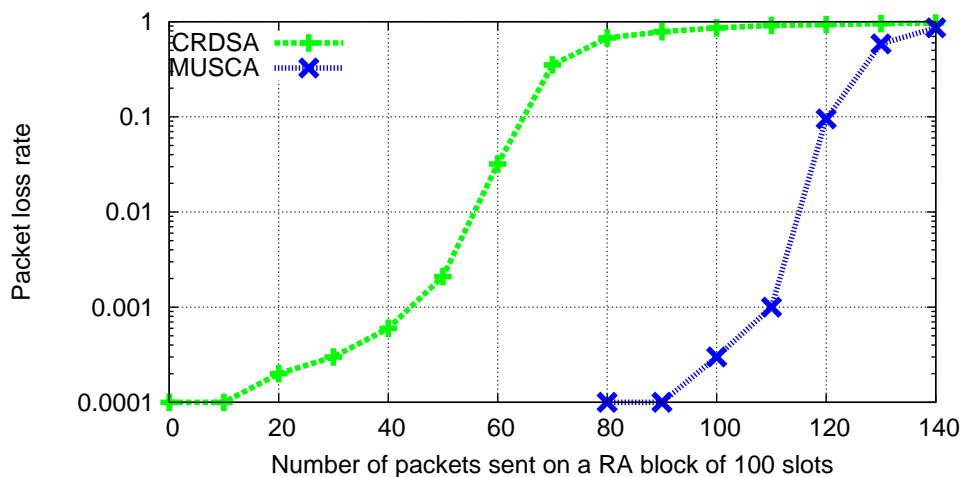

Figure 3: Packet loss rate and number of packets sent on a RA block of 100 slots with CRDSA and MuSCA at 5dB

4.5 Topology, traffic and hypothesis

We consider two nodes in NS-2. The first node represents the set of ST and the second node acts as the satellite gateway. The PCA module presented in Section 3 is introduced from the ST to the satellite gateway. We present in Table 2 the different traffic types considered in the following sections. The size of the IP datagrams is 1,500 bytes, and the queue at the transmitter is large enough to prevent overflowing.

Various TCP options could have been considered as they might have an impact on the resulting traffic load on the return link. As an example, web browsing results in up to 6 TCP flows for a single page and the acknowledgments that would be sent from the home user to the server would be synchronized for the first objects downloaded. This synchronization would result in a momentarily high traffic load on the return link that might be hard to handle and induce loss of acknowledgments. In this context, Delayed Acknowledgments or TCP Pacing would break that synchronization and then reduce the potential loss of crucial acknowledgments. We do not consider these options in the rest of the paper, use default implementations of TCP. We however note that improvements at the transport layer may result in slight performance variations. We note that considering a single access point is

Table 2: Traffic generation

	Case A Variable number of TCP bull transfer	**Case B**	**Case C**	**Case D** HTTP traffic Packmime
Data from ST to gateway	✓	✓	✓	
Data from gateway to ST				✓
TCP version	CUBIC	CUBIC	CUBIC	NewReno
Flow size (request size for Packmime)	∞	$30\,\text{kB}$	$4.5\,\text{kB}$	$\in [150; 650]\,\text{B}$
Start time [s]	0	10	2	0
End time [s]	20	N/A	N/A	20

a simplification, but it will not overly affect the qualitative results derived from our simulation. We also did not dive into the lower layer parameterization as we are interested in a high level view of the performance at the transport layer.

5 Evaluating the performance of random access methods for data traffic

In this section, we assess the benefits of utilizing random access methods in DVB-RCS2 for transmitting data.

5.1 Experimental scenario

We consider the parameters detailed in Table 1, *i.e.*, there are $(F_D + F_R) \times N_S = 100 \times 40 = 4,000$ slots per frame. We also introduce the Single terminal amplifier limitation detailed in Section 4.2, which limits the maximum number of slots per TCP session to 13 for random access methods and 40 for dedicated access method. Moreover, the capacity is fairly shared between the N_U ST using a dedicated access method, therefore the maximum number of slots per TCP sessions in this case is: $\min((F_D + F_R) \times N_S/N_U, N_S) = N_S \times \min((F_D + F_R)/N_U, 1)$.

When $N_U \leq (F_D + F_R)$ (*i.e.*, with our parameters, when $N_U \leq 100$), a TCP session can transmit $\lfloor N_S/N_{block} \rfloor \times N_{data} = 40/1 \times 920 = 36,800$ bits per frame with a dedicated access, or $\lfloor 40/3 \rfloor \times 594 = 7,722$ bits with MuSCA as random access. With dedicated access methods, each TCP session can transmit more data per frame when the load is low. When the load increases: (1) with dedicated access methods, the goodput of each TCP session decreases and (2) with random access methods, the goodput of each TCP session remains the same, but there might be collisions at the MAC layer, resulting in a lower goodput.

Considering phenomena (1) and (2), we now evaluate which access scheme allows each TCP session to have a better goodput when the load increases. We consider traffic case A of Table 2: we measure the range of network loads for which makes the choice of random access methods is more suitable to transmit data than dedicated access methods.

5.2 Throughput and datagram loss rate

We show in Figure 4 the average number of datagrams sent per TCP sessions. Additionally, Figure 5 gives the error probability by considering the number of datagrams dropped at the gateway level and the number of datagrams successfully transmitted. Finally, to better assess the impact of the datagrams errors on the transmission efficiency, we show in Figure 6 the efficiency determined by the ratio of the average number of transmitted datagrams to the maximum number of datagrams that could have been transmitted without errors.
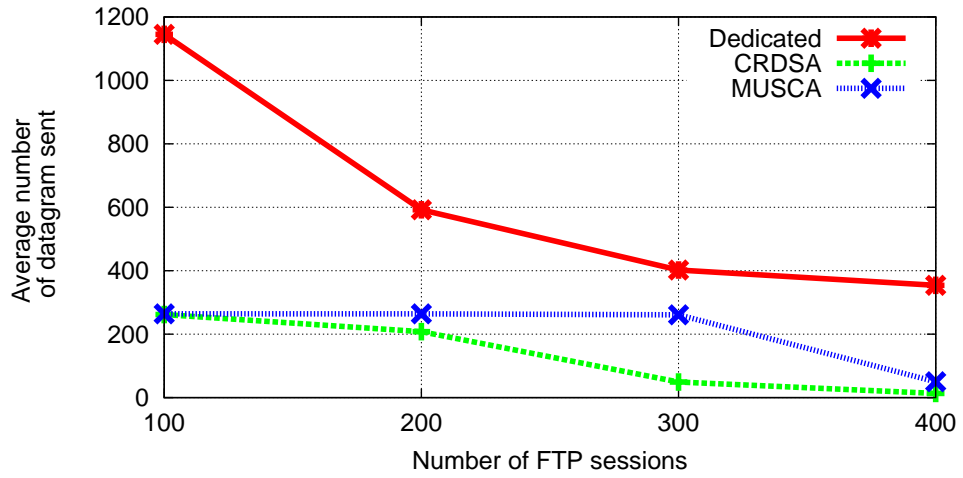
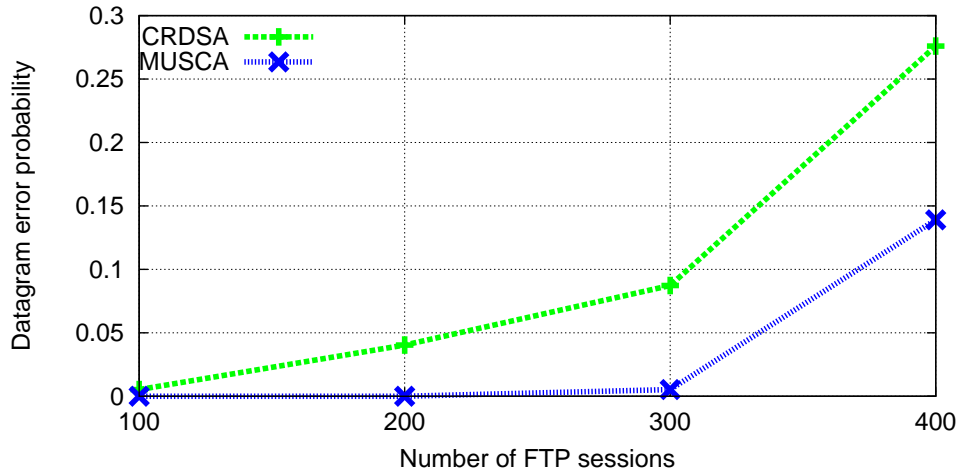Figure 4: Average number of datagrams sent per TCP session in 20 s

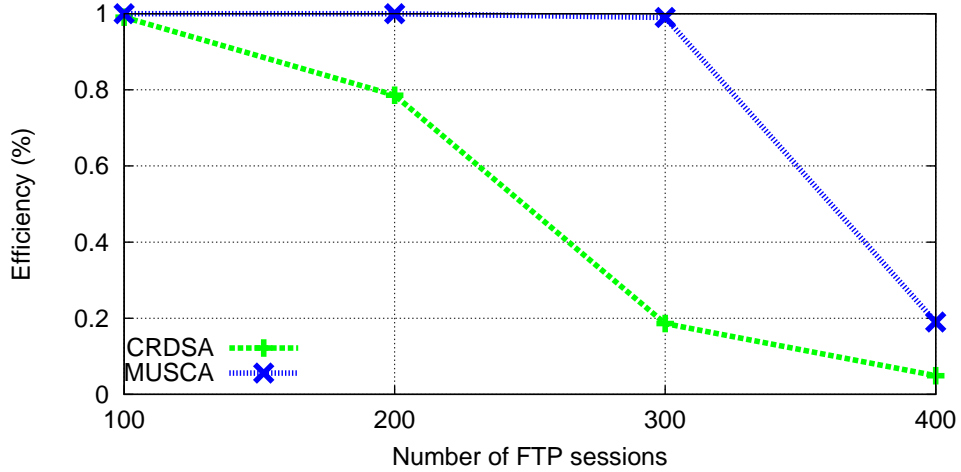Figure 5: Average number of datagrams lost per TCP sessions in 20 s

Figure 6: Transmission efficiency, the ratio between the average number of datagrams transmitted over the maximum number of datagrams that can be transmitted in 20 s per FTP session

From Figure 5, we see that datagram errors appear from $N = 100$ for CRDSA and $N = 300$ for MuSCA. This results in an efficiency (ratio between the average number of datagrams transmitted over the maximum number of datagrams that can be transmitted in 20 s per FTP session) of the transmission that decrease when the number of FTP sessions increases as illustrated in Figure 6. However, Figure 4 illustrates that when $N \leq 300$, one TCP session transmits more datagrams with dedicated access methods than with any random access methods. Moreover, when the load increases, the datagram errors increases consequently with random access methods and, as a result, the average number of datagrams sent per TCP session decreases. As an example, with MuSCA as access method, one TCP session transmits on average 50 datagrams, whereas with dedicated access, one TCP session transmits on average more than 350 datagrams.

5.3 Discussion

We observed in our NS-2 simulations that, even though the throughput of each TCP session decreases when the load of the network increases, dedicated access methods enable the transmission of more data than random access schemes which lose a large number of datagrams: phenomena (1) of Section 5.1 is not as bad as phenomena (2).

Indeed, we earlier derived that the maximum number of slots available per frame per TCP session is defined by $N_S \times \min((F_D + F_R)/N_U, 1)$. When $N_U \geq (F_R + F_D)$, a TCP session can transmit $(F_R + F_D) \times N_S/N_U \times N_{data}^{DE}$ bits per frame with a dedicated access, and $\lfloor N_S/N_{block} \rfloor \times N_{data}^{RA}$ with MuSCA as a random access method. For a future random access to be viable, the number of users, $N_U' \geq (F_R + F_D)$, that enables it to transmit more data should verify Equation (2).

$$(F_R + F_D) \times N_S/N_U' \times N_{data}^{DE} \leq \lfloor N_S/N_{block} \rfloor \times N_{data}^{RA} \tag{1}$$

$$N_U' \geq \frac{(F_R + F_D) \times N_S/N_U' \times N_{data}^{DE}}{\lfloor N_S/N_{block} \rfloor \times N_{data}^{RA}} \tag{2}$$

Figure 7 illustrates the performance that a random access method should have to be more efficient than dedicated access methods when the load on the network is large. We denote by N_{MaxRA} the number of users from which the random access methods starts to introduce errors (*i.e.*, $N_{MaxRA} = 300$ with MuSCA).

With our parameters, the minimal desirable N_U' is such that $N_U' \geq 477$ applying (2). With MuSCA as a random access method, these N_U' users would be equally spread among the 40 RA blocks. There
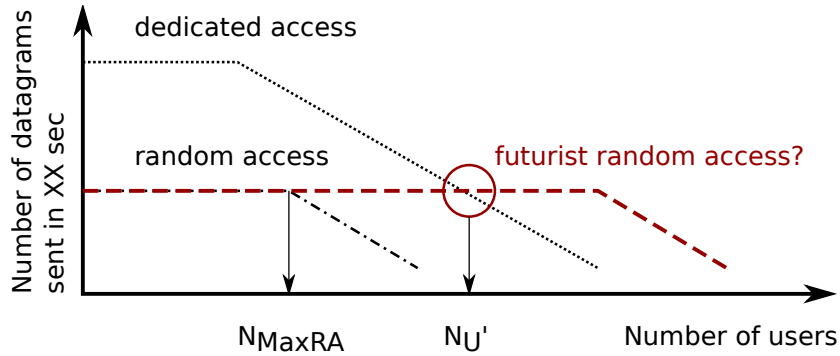
Figure 7: Illustration of N'_U

would be $N'_U \times 13/40 = 477 \times 13/40 = 155$ users per RA block, which MuSCA can not carry, as illustrated in Figure 3.

As far as we know, there is no random access method that can carry traffic and verify (2). However, this equation can help to assess the load from which it is interesting to transmit some data with random access methods.

Both NS-2 simulations and mathematical expressions illustrate that the transmission of data is more efficient with dedicated access methods, as random access methods enable to transmit less data on one given frame and errors might occur. However, the next section looks more closely at detailed performance of one given flow in order to explain the rational of introducing random access methods to carry short data flows.

6 Transmission times of short flows

Section 5 showed that data transmission is more efficient with dedicated access methods. However, considering that (1) there is a connection delay introduced by dedicated access methods, (2) there is an important proportion of short flows in the Internet (measured in [17, 24]), we now study the benefits that random access methods can provide in terms of transmission delay for short flows when there are no errors (we focus on cases when $N < N_{MaxRA}$ in Figure 7).

6.1 TCP sessions

We consider the conjoint generation of traffic cases A and B of Table 2. In Figure 8, we plot the evolution of the TCP segment sequence for one typical flow decoded at the receiver side with 150 TCP sessions. We consider flows that do not lose datagrams when random access methods are introduced. We can see the progression of the congestion window of TCP in the slow start phase, with CWND and RTT presented in the figure. Overall, this figure illustrates that the RTT needed for the connection when dedicated access is involved delays the transmission of the first datagrams. We can also see that the time needed to transmit two datagrams is smaller with dedicated access (denoted $T2$) than with random access (denoted $T1$). As a result, with dedicated access, the progression of the congestion window is faster, but starts later.

The amount of useful bits that can be sent on each slot is 597 with CRDSA and 594 with MuSCA: when there are no losses, which is the case in the results presented in Figure 8, the performance for a given flow similar.

We present the time needed to transmit $30\,\mathrm{kB}$ (traffic case B of Table 2) in Table 3. When there are 200 TCP sessions, both CRDSA and MuSCA transmit the $30\,\mathrm{kB}$ faster by $90\,\mathrm{ms}$ than dedicated access. We confirm that when there are more TCP sessions the time needed to transmit $30\,\mathrm{kB}$, with dedicated access methods, is larger, *i.e.*, resulting in lower throughput for each user. Conversely, the transmission of those 20 datagrams is faster with random access methods (when there are no errors).
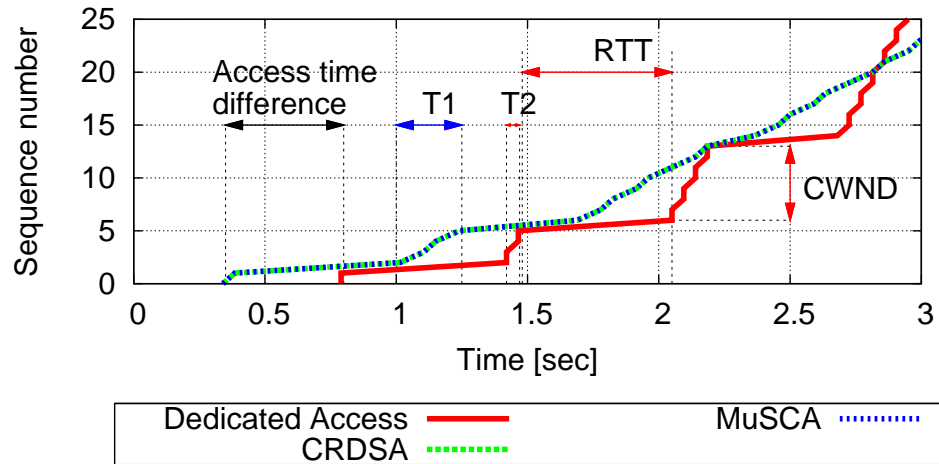
Figure 8: Evolution of TCP segment sequence number reception

Table 3: Transmission times of 30 kB

Number of competing flows	Access method	Reception date	
		first packet	last packet
150	Dedicated	10.825	12.715
150	CRDSA	10.375	12.67
150	MuSCA	10.375	12.67
200	Dedicated	10.825	12.76
200	CRDSA	10.375	12.67
200	MuSCA	10.375	12.67

We propose in the next section to validate this interpretation by considering a more realistic traffic model.

6.2 HTTP traffic with Packmime

Packmime [11] is an NS-2 module that models HTTP traffic. It is controlled by a rate parameter, *i.e.*, the average number of new connections that start each second. This module enables us to model clients (ST) that send requests to the servers (satellite gateway). We consider the traffic case D of Table 2.

We let Packmime generate 2,000 requests and observe the flows' performance with dedicated and random access methods (the performance of MuSCA and CRDSA are identical). The size of the requests generated by the clients is within [150;650] bytes and the rate is set to 500. We define the transmission time as the time between the transmission of the first request (SYN/SYN ACK of 40 bytes) from the client and the reception of the last one at the server. The results are summarised in Table 4 which show the minimum, median and maximum requests transmission times.

Table 4: HTTP request transmission times

Access method	Transmission time [s]		
	Minimum	Median	Maximum
Dedicated access	1.33	1.40	1.55
Random access	0.88	0.91	1.10

This confirms that the transmission of short requests is faster with random access methods.

6.3 Short flows with errors

The conclusions from the previous section must be adjusted with evaluations on the impact of network load (and resulting collisions) on the transmission time. We now consider scenarios where the network load is too high for random access methods to transmit data without error ($N > N_{MaxRA}$). We measure the maximum transmission times of short flows when error events occur. The goal of this section is also to determine the minimum delay introduced by retransmissions depending on the number of datagrams and the state of the network.

We consider that one user transmits D datagrams. We denote by $P_{err}(N)$ the probability to lose a datagram with N users (from Figure 5), and by $\Delta = \{d_1; d_2; \dots; d_D\}$ the set of datagrams of one flow. $P(r_{d_i} = R)$ is the probability that the datagram d_i is retransmitted R times and is determined by (4).

$$\forall R \in \mathbb{N}, \forall N \in \mathbb{N}, \forall d_i \in \Delta, \tag{3}$$

$$P(r_{d_i} = R) = (1 - P_{err}(N)) \times P_{err}(N)^R \tag{4}$$

We determine the probability to have at least $\hat{R}$ retransmissions for one datagram. As one retransmission increases the delay by at least one RTT, we propose a simple lower bound of the time needed to transmit a flow composed of several datagrams. As illustrated in Figure 9, we focus on the first datagrams of one TCP session, we do not consider the congestion avoidance phase. We compute a "lower" bound for the supplementary delay introduced by loss events.

$$P(\hat{R}) = \sum_{k=0}^{D-1} \binom{D}{k} P(r < R)^k \times P(r = R)^{D-k} \tag{5}$$

$$P(\hat{R}) = \sum_{k=0}^{D-1} \binom{D}{k} \left(\sum_{i=0}^{R-1} P(r = i) \right)^k \times P(r = R)^{D-k} \tag{6}$$

Using (4) and (6), we derive numerical evaluations in Table 5, using the error probability and resulting delays, depending on the number of users from Figure 5.

Table 5: Retransmission probabilities

Number	Retransmissions		P($\hat{\text{R}}$)	
of users	number	min. add. delay	CRDSA	MuSCA
200	1	0.500	0.11148	0
200	2	1.000	0.00469	0
200	3	1.500	0.00019	0
300	1	0.5	0.21701	0.01573
300	2	1.0	0.02067	0.00008
300	3	1.5	0.00182	4.44e-7
400	1	0.5	0.40893	0.30487
400	2	1.0	0.14980	0.04882
400	3	1.5	0.04445	0.00692
400	4	2.0	0.01251	0.00096

When $N \geq N_{MaxRA}$, the probability of collision events, inducing retransmissions, can not be neglected. During the transmission of 3 datagrams, the probability for one of those to be lost can be up to 22% with CRDSA and 300 TCP sessions and provoke at least an additional 500 ms in transmission time. We showed in Figure 8 that the transmission of 3 datagrams without loss is faster with random access methods: when a loss event occurs, the whole benefits provided by random access methods is lost. We conjointly generated the traffic cases A and case C of Table 2. The starting time is different from the case B, since we consider that there might be losses and retransmissions in this section, and we did not want to bias the results by limiting the maximum transmission time. When there are 200 TCP sessions (resp. 300 TCP sessions), with CRDSA (resp. MuSCA) as random access method, over 250 runs, the maximum transmission time is 4.325 s (resp. 4.145 s).

7 Discussion: mixing random and dedicated access methods

We verified in Section 6 that the transmission of the first packet of a flow was indeed faster with random access methods under low network loads (*i.e.*, $N < N_{MaxRA}$). These results are consistent with those presented in [5], where the performance of TCP over DVB-RCS2 and random access schemes are assessed. We cannot compare our numerical values, as we considered different metrics and they only look at normalized TCP goodput. The qualitative trend is the same.

Figure 9 shows that, depending on the traffic load, the benefits provided by random access vary. When the load increases, the capacity is shared among the users with dedicated access and the time needed to transmit a certain amount of datagrams increase. We quantified the situations when datagrams errors introduce a delay, cancelling out the benefits that random access methods may provide.

We show, in Figure 10, where a switch from random to dedicated access could be introduced to improve the transmission of both short and long flows. The idea is to enable a faster transmission for the first packet of the TCP sessions by adapting the choice of the access method. To increase the transmission of the first datagrams, they must be transmitted over random access methods. This should however be done only when the network load is low enough, that is, in the congestion range for which random access methods can guarantee the error-free delivery—or recovery—of the transmitted packets. Obviously, one issue would be to predict how the traffic load for the next frame, but we believe that in 45 ms (a super frame duration) the traffic would not have time to drastically change. In order to guarantee room for flows transmitting over the random access method, the traffic over the dedicated access methods could be shaped so that the peaks of incoming of traffic are reduced; this can be achieved by using similar techniques than those presented in [15]. The results presented in the article let us argue that, for a given TCP session, starting with random access before switching
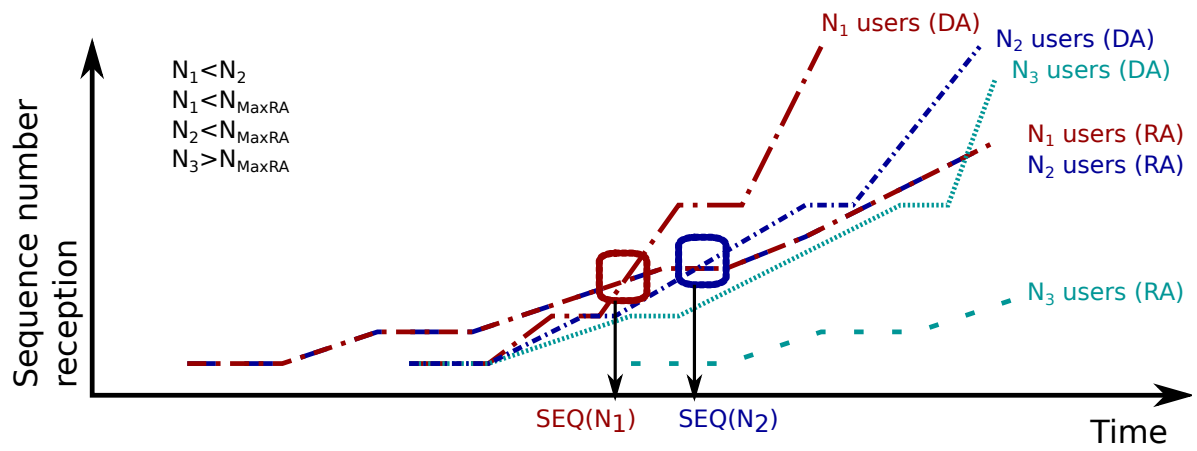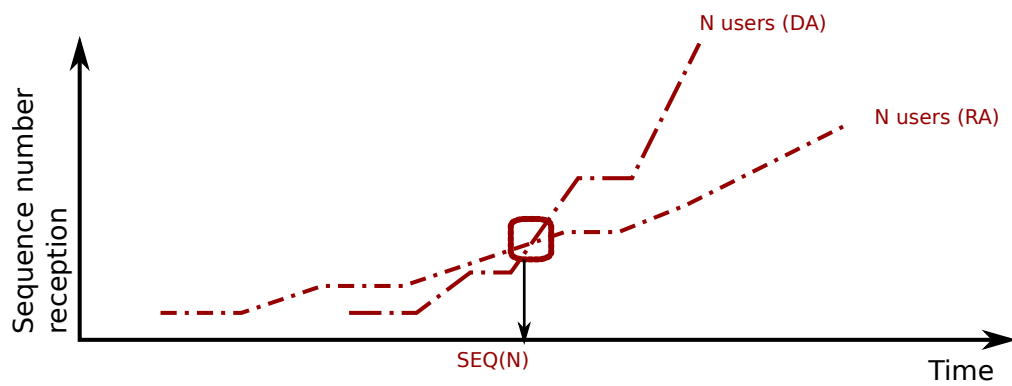
Figure 9: Datagrams errors and short flows

Figure 10: Switch from random to dedicated access

to dedicated access can be benifical. This would improve end-user experience of the service as well as spectrum efficiency by allowing the first data packets to be sent, earlier, on RA blocks rather than waiting for a dedicated reservation. This is in line with [21], which presents analytical derivations of the benefits of such switching for a given application. Their benefits require further validation through simulations. This idea is also mentioned in IP Over Satellite (IPOS) standards. We argue for the integration of a similar strategy in the current DVB-RCS2 standards.

8   Conclusion

The current DVB-RCS2 standards do not mandate a specific channel access strategy for the return satellite link (transmission of data from home users to satellite gateways). In this article, we compared the impact of both dedicated and random access methods on the performance of TCP. We note that our simulations used a few simplifying assumptions with respect to lower layers schemes and the topology. While this might impact the accuracy of quantitative results, the qualitative trends were clear. We presented an NS-2 module, PCA, that emulates the access to the DVB-RCS2 return link. PCA can be directly reused for any other scenarios where satellite channel access methods are involved. We showed that the transmission of data is more efficient with dedicated access methods, as random access methods enable to transmit less data per frame and errors might occur. However, we also found that the transmission of short flows is faster with random access methods. Service providers can benefit from reduced page loading time. For example, Amazon found that it increased revenue by 1% for every 100 ms in page loading time [7]. We believe that, in order to provide a good web experience, random access methods and a switch between both schemes should be considered as part of the DVB-RCS2 specification effort.

References

[1] Second generation dvb interactive satellite system (dvb-rcs2); part 3: Higher layers satellite specification. In *Digital Video Broadcasting (DVB) TS 101 545-3*.

[2] Second generation DVB interactive satellite system; part 1: Overview and system level specification. In *Digital Video Broadcasting (DVB) TS 101 545-1*.

[3] Second generation DVB interactive satellite system; part 2: Lower layers for satellite standard. In *Digital Video Broadcasting (DVB) ETSI EN 101 545-2 V1.1.1*.

[4] Norman Abramson. The ALOHA system: another alternative for computer communications. AFIPS '70 (Fall), 1970.

[5] M. Bacco, A. Gotta, C. Roseti, and F. Zampognaro. A study on TCP error recovery interaction with Random Access satellite schemes. In *Advanced Satellite Multimedia Systems Conference and the 13th Signal Processing for Space Communications Workshop (ASMS/SPSC), 2014 7th*, 2014.

[6] F. Belli, M. Luglio, C. Roseti, and F. Zampognaro. Evaluation of TCP performance over emulated DVB-RCS scenario with multiple RCSTs. In *International Workshop on Satellite and Space Communications (IWSSC)*, 2009.

[7] Joshua Bixby. Top ecommerce sites are 22% slower than they were last year. 2013.

[8] Barłomiej Błaszczyszyn, Paul Mühlethaler, and Yasser Toor. Stochastic analysis of aloha in vehicular ad hoc networks. *Annals of Telecommunications*, 2013.

[9] Huyen Chi Bui, Jérôme Lacan, and Marie-Laure Boucheret. Access methods based on physical-layer network coding. In *Ph.D. thesis*, ISAE, 2012.

[10] Huyen Chi Bui, Jérôme Lacan, and Marie-Laure Boucheret. An enhanced multiple random access scheme for satellite communications. In *Wireless Telecommunications Symposium (WTS)*, 2012.

[11] Jin Cao, W.S. Cleveland, Yuan Gao, K. Jeffay, F.D. Smith, and M. Weigle. Stochastic models for generating synthetic http source traffic. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, 2004.

[12] E. Casini, R. De Gaudenzi, and Od.R. Herrero. Contention resolution diversity slotted ALOHA (CRDSA): An enhanced random access schemefor satellite access packet networks. In *IEEE Transactions on Wireless Communications*, number 4, april 2007.

[13] N. Celandroni, F. Davoli, E. Ferro, and A. Gotta. Employing contention resolution random access schemes for elastic traffic on satellite channels. In *18th Ka and Broadband Communications Navigation and Earth Observation Conference*, 2012.

[14] N. Celandroni and R. Secchi. Suitability of dama and contention-based satellite access schemes for tcp traffic in mobile dvb-rcs. *IEEE Transactions on Vehicular Technology*, 2009.

[15] Eduardo Andrés Celis Muñoz, Fabrice Le Denmat, Arnaud Morin, and Xavier Lagrange. Multimedia content delivery trigger in a mobile network to reduce the peak load. *Annals of Telecommunications*, 2014.

[16] G. Choudhury and S. Rappaport. Diversity ALOHA–a random access scheme for satellite communications. In *IEEE Transactions on Communications*, 1983.

[17] D. Ciullo, M. Mellia, and M. Meo. Two schemes to reduce latency in short lived TCP flows. In *Communications Letters, IEEE*, volume 13, pages 806 –808, october 2009.

[18] David Espes, Xavier Lagrange, and Luis Surez. A cross-layer mac and routing protocol based on slotted aloha for wireless sensor networks. *Annals of telecommunications*, 2015.

[19] Thierry Gayraud, Lionel Bertaux, and Pascal Berthou. A NS-2 Simulation Model of DVB-S2/RCS Satellite Network. In *15th Kaband Conference*, page 1, Italy, September 2009. 09702 09702.

[20] O. Herrero, R. Gaudenzi, and J.L.P. Vidal. Design guidelines for advanced random access protocols. In *International Conference on Satellite and Space Communications*, 2012.

[21] C. Kissling and A. Munari. On the Integration of Random Access and DAMA Channels for the Return Link of Satellite Networks. In *IEEE International Conference on Communications (ICC)*, june 2013.

[22] Nicolas Kuhn, Huyen Chi Bui, Jérôme Lacan, José Radzik, and Emmanuel Lochin. On the benefits of random access methods on TCP performance over DVB-RCS2. In *ACM MobiCom Workshop on Lowest Cost Denominator Networking for Universal Access (LCDNet)*, 2013.

[23] Nicolas Kuhn, Olivier Mehani, Huyen-Chi Bui, Jérôme Lacan, José Radzik, and Emmanuel Lochin. Physical Channel Access (PCA): Time and Frequency Access Methods Simulation in NS-2. In *5th International Conference on Personal Satellite Services (PSATS)*, 2013.

[24] C. Labovitz, S. Iekel-Johnson, D. McPherson, J. Oberheide, F. Jahanian, and M. Karir. Atlas internet observatory annual report. In *47th NANOG*, 2009.

[25] Emmanuel Lochin, Tanguy Pérennou, and Laurent Dairaine. When should I use network emulation? *Annals of Telecommunications*, 67:247–255, 2012.

[26] M. Luglio, F. Zampognaro, T. Morell, and F. Vieira. Joint DAMA-TCP protocol optimization through multiple cross layer interactions in DVB RCS scenario. In *International Workshop on Satellite and Space Communications (IWSSC)*, 2007.

[27] O3b Networks and Sofrecom. Why latency matters to mobile backhaul. Technical report, O3b Networks, Jersey, Channel Islands, April 2013.

[28] C. Roseti and E. Kristiansen. TCP behaviour in a DVB-RCS environment. In *International Communications Satellite Systems Conferences (ICSSC)*, jun. 2006.

[29] R. Secchi. DVB-RCS(2) for ns-2. Technical report, University of Aberdeen, 2011.

[30] A. Tambuwal, R. Secchi, and G. Fairhurst. Exploration of random access in DVB-RCS. In *Post-Graduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting (PGNET)*, 2011.