Optimal batch scheduling in DVB-S2 satellite networks

G. T. Peeters, B. Van Houdt and C. Blondia
University of Antwerp, Department of Mathematics and Computer Science,
Performance Analysis of Telecommunication Systems Research Group - IBBT,
Middelheimlaan 1, B-2020 Antwerp, Belgium

***Abstract*—In this paper we present a new theoretical model to assess the performance of a class of batch scheduling orders in a forward DVB-S2 satellite link. The scheduling order in a DVB-S2 link will determine the order in which IP packets are transmitted to the receivers and thus determines the amount of required padding and sub-optimal transmissions. The validity of the model is investigated by various simulation runs which show a very close agreement with the theoretical model. Using this model we also identify the optimal scheduling order within this class of orders, given that some mild and realistic conditions on the modulation and coding schemes apply.**

I. Introduction

In 2004, a new communication standard, concerning the forward link on GEO satellites, DVB-S2, was proposed. A series of features have been introduced over the older standard, DVB-S [1], such as improved modulation and coding possibilities, a generic stream (GS) interface, as well as Adaptive Coding and Modulation (ACM) [2].

The GS interface introduced by DVB-S2, allows IP data to be sent more efficiently than existing MPE/ULE over MPEG transport stream (TS) encapsulation solutions [3]. Using this interface, it is now possible to place IP packets directly in the DVB-S2 frame structure, omitting any MPEG TS signaling and requirements. At the time of writing, a standardized protocol for IP over GS has not yet been finalized; one remaining issue concerns the allowed fragmentation options. Some possible approaches have been proposed in [4], with the most restrictive being Classical Consecutive Fragmentation (CCF). The essence of this scheme is that when a packet is started but does not fit in the current frame, it must continue at the start of the next frame. Alternatively, the entire packet transmission can also be delayed until the next frame, padding the remainder of the current frame. Within this paper, we will assume CCF, since this is the most restrictive and therefore the most "compatible" of all proposed schemes; other schemes, perhaps allowing more efficient scheduling, require more complex reassembly routines.

Another DVB-S2 feature, ACM, allows adapting the modulation and coding of the transmission on a single link on a per frame basis, as opposed to DVB-S where these parameters were fixed per link basis. Each combination of a specific modulation and coding scheme, referred to as a MODCOD, has a certain spectral efficiency that expresses the rate at which *data* can be transmitted. While a high spectral efficiency thus indicates the possibility to achieve high data rates, it also requires a better signal-to-noise ratio (SNR) from the receiver. For a single receiver, this means that, given an SNR at a certain time (mostly determined by its weather conditions), there always exists a single optimal MODCOD at which data can be received: that is, higher (more efficient) MODCOD transmissions require a better SNR, while the reception of lower MODCOD transmissions is possible (as the required SNR is met), but implies a waste of effective bandwidth.

DVB-S2 frames have a semi-fixed payload size, meaning that some form of IP packet fragmentation and/or padding is required. As the same MODCOD must be used within a DVB-S2 frame and we wish to support CCF fragmentation, the scheduling order has a profound impact on the link utilization as it must schedule various packets destined for different receivers, having different receiving conditions (i.e., SNRs) and thus different optimal MODCODs. We consider a class of scheduling orders compliant with the CCF functionality and develop a theoretical model to calculate the expected link utilization loss due to padding and sub-optimal transmissions.

DVB-S2 is not the first communication standard incorporating dynamic modulation/coding techniques; for instance the wireless communication standard 802.11b provides an adaptive rate mechanism, setting the data rate per frame, according to the SNRs of the individual nodes. However, unlike in 802.11b where each frame addresses a single (or broadcast) station, IP over DVB-S2 GS allows to place multiple IP packets into a single frame. One key issue is that, to take maximum advantage of ACM, using CCF, a single DVB-S2 frame, which has a constant MODCOD, can contain IP packets for receivers with different optimal MODCODs.

The paper is organized as follows. Section II provides a detailed problem description as well as the model assumptions. A theoretical analysis of the efficiency of a class of batch scheduling algorithms under these assumptions is provided in Section III-A. This analysis also allows to identify the optimal scheduling within this class (see Section III-B), giving rise to some simple engineering rules to optimize the link utilization. Finally, a series of simulation runs are presented in Section III-C to demonstrate the validity of the theoretical model.

II. Problem statement, model description

The DVB-S2 standard divides the (one-to-many) forward link into frames, where each frame can be characterized by three parameters. First, we have that the effective payload size,

1930-529X/07/$25.00 © 2007 IEEE
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE GLOBECOM 2007 proceedings.

i.e. the number of *data* bits, depends on the frame's coding parameter. For instance, a coding parameter of $2/3$ indicates that for every 2 data bits, there is one extra coding bit. The total frame size, which combines this data with additional coding bits, is fixed, and will be referred to as *raw* bits. Finally at the physical layer, the frame length is expressed in symbols and therefore varies in time, where the modulation parameter determines the length (as this parameter reflects the number of raw bits that can be carried per symbol). For example, if a symbol carries 4 instead of 2 raw bits, the frame duration is halved.

When scheduling IP data in DVB-S2 frames, we note that each IP packet has a destination with an associated maximal MODCOD. Throughout this paper we will state that an IP packet with (maximal) MODCOD i belongs to class-i. Given the frame length (in data bits) and packet size variability, IP packet boundaries do generally not match with frame boundaries, preventing MODCOD switches to align with IP packets. Since the MODCOD is fixed for the duration of a frame, CCF limits our options in case we wish to schedule (a part of) two packets with different MODCODs within a single frame. Consider the following two scenarios:

1) The second packet has a higher (optimal) MODCOD than the current frame. In this case the packet can be transmitted (sub-optimally) in the current frame.
2) The second packet has a lower MODCOD, which prevents the transmission in the current frame; so, we have to pad the remainder of the current frame, and transmit the second packet (at the start of) the next frame with a sufficiently low MODCOD.

As we see, transitions from one class of IP packets to another come at a cost either due to sub-optimal transmissions or due to padding. The goal of this paper is to find batch scheduling orders that minimize this additional cost.

A. Model assumptions

Consider all MODCODs in a scheduling setup. MODCOD i can then be characterized by considering a frame, transmitted in MODCOD i, where two parameters are relevant: the number of data bits per frame (defined by the COD), denoted by e_i, and the number of raw bits per symbol (defined by the MOD), expressed by s_i. With this information, combined with the constant frame size T (in raw bits per frame), we can calculate the cost for a data bit, in symbols, for any MODCOD.

A MODCOD i is said to have higher (or equal) spectral efficiency than MODCOD j if and only if $e_i s_i \geq e_j s_j$, that is, if its use results in more *data* bits per time interval. We assume that the MODCODs are ordered from low to high, i.e., $e_1 s_1 \leq e_2 s_2 \leq \ldots \leq e_n s_n$. The row of modulations $s_1, \ldots, s_n$ can be constant (same MOD for all MODCODs), increasing (meaning a higher MODCOD implies a higher MOD), decreasing or none of the above (see Table I and Figure 1).

Two additional assumptions to set up the theoretical model are introduced next.

MODCOD	i	e_i	s_i	$e_i s_i$
QPSK 1/4	1	16008	2	32016
QPSK 1/3	2	21408	2	42816
QPSK 2/5	3	25728	2	51456
QPSK 1/2	4	32208	2	64416
QPSK 3/5	5	38688	2	77376
QPSK 2/3	6	43040	2	86080
QPSK 3/4	7	48408	2	96816
QPSK 4/5	8	51648	2	103296
QPSK 5/6	9	53840	2	107680
QPSK 8/9	10	57472	2	114944
8PSK 3/5	11	38688	3	116064
QPSK 9/10	12	58192	2	116384
8PSK 2/3	13	43040	3	129120
8PSK 3/4	14	48408	3	145224
8PSK 5/6	15	53840	3	161520
16APSK 2/3	16	43040	4	172160
8PSK 8/9	17	57472	3	172416
8PSK 9/10	18	58192	3	174576
16APSK 3/4	19	48408	4	193632
16APSK 4/5	20	51648	4	206592
16APSK 5/6	21	53840	4	215360
16APSK 8/9	22	57472	4	229888
16APSK 9/10	23	58192	4	232768
32APSK 3/4	24	48408	5	242040
32APSK 4/5	25	51648	5	258240
32APSK 5/6	26	53840	5	269200
32APSK 8/9	27	57472	5	287360
32APSK 9/10	28	58192	5	290960

TABLE I
ALL COMBINATIONS OF MODULATION AND CODING WITH THEIR CORRESPONDING e_i, s_i AND $e_i s_i$, SUPPORTED BY THE DVB-S2 STANDARD, FOR A FRAME LENGTH OF $T = 64800$ RAW BITS/FRAME.

a) Batch processing: The IP packets are assumed to arrive in large batches, sorted per MODCOD class. The scheduler is assumed to process each batch separately. Once the entire batch is processed, the next batch is to be scheduled. A scheduling result for a single batch indicates in which order the n queues are visited, as well as the chosen MODCODs for each frame. Our objective is to minimize the overhead among all orders, assuming that the scheduling order is chosen identical for all batches.

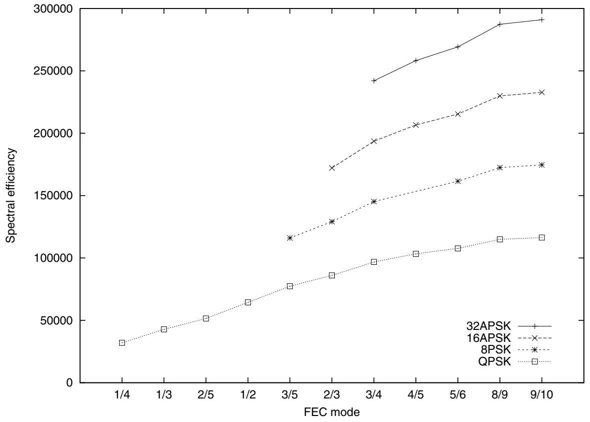

Fig. 1. Illustration of the spectral efficiency $e_i s_i$ for all DVB-S2 MODCODs.

b) Batch size: We assume that we are dealing with large batches, consisting of many IP packets of each class. Also, noting that the amount of data bits per frame e_i is large (see table I for real world examples), standard renewal theory results [5] indicate that the time at which the queue i runs out of IP packets, may be considered as random within a frame. Allowing us to establish results that are independent of the packet length distribution.

If we consider all packets with the same MODCOD, we can either schedule them consecutive, or we can allow placing one or more packets from other classes in between. As switches from one class to another involve some overhead, we can intuitively see that the consecutive execution is optimal. Furthermore, this class of scheduling orders ensures that no packet resequencing occurs.

III. Efficiency analysis

To express the efficiency of a scheduling order S, we will compute the amount of symbols used to transmit all IP packets during a single scheduling batch, minus the cost needed if all IP packets were transmitted entirely in their optimal MODCOD; we will refer to this difference as the scheduling overhead E_S. Using this overhead we will be able to compare possible scheduling orders, and thus choose the optimal order. Note that we do not consider protocol overhead.

We start with a few general equalities, describing this overhead, due to transitions from one MODCOD to another, independent of the scheduling strategy.

A. $L \rightarrow H$ and $H \rightarrow L$ transitions

A transition between two classes of IP packets generally involves some overhead. We distinguish between two types of transitions: $L \rightarrow H$ (Low to High MODCOD) and $H \rightarrow L$ (High to Low MODCOD). The derived expressions also apply to the transitions between the last and first class of two consecutive batches.

a) $L \rightarrow H$ transition: Assume that while filling a MODCOD i frame we run out of MODCOD i packets. In such case we will fill the remainder of the frame with MODCOD $j > i$ data. As argued below, there is a cost associated with this MODCOD switch (in symbols):

$$\frac{e_i}{2}\left(\frac{T}{e_is_i}-\frac{T}{e_js_j}\right)=\frac{T}{2}\left(1-\frac{e_is_i}{e_js_j}\right)\frac{1}{s_i}. \qquad (1)$$

Indeed, on average we need to add $e_i/2$ bits of MODCOD j data. T equals the number of raw bits per frame, s_i is the number of raw bits per symbol; hence, T/s_i equals the number of symbols per frame. As e_i data bits fit into a MODCOD i frame, we find that $T/(e_is_i)$ symbols are required per data bit in a MODCOD i frame. Thus, transmitting $e_i/2$ bits takes $e_i/2\ T/(e_is_i)$ symbols in MODCOD i, while ideally only $e_i/2\ T/(e_js_j)$ are needed (in MODCOD j).

b) $H \rightarrow L$ transition: When we run out of MODCOD j packets while filling a frame and MODCOD $i < j$ data is to follow, there are two options: (i) either we *switch* all the data in the frame to MODCOD i, (ii) or we *pad* the remainder of the frame. We start by establishing a condition that indicates which option is best, provided that q MODCOD j data bits are present when running out of packets. Option (i) implies that we need $qT/(e_is_i)$ symbols to transmit these q bits, whereas option (ii) requires a total of T/s_j symbols, therefore we will pad the remainder of the frame whenever

$$\frac{T}{s_j} < q\frac{T}{e_is_i} \Leftrightarrow q > \frac{e_is_i}{s_j},$$

otherwise, we use these q bits to start a MODCOD i frame. Notice, in some cases these q bits, when transmitted in MODCOD i, may occupy more than one frame. For example, one and a half 16APSK1/4 frames require less symbols than a QPSK3/4 frame. Thus, on average, we will switch MODCODs with probability $p = e_is_i/(e_js_j)$ (as q is assumed to be uniform in the interval $(0, e_j)$) and pad with probability $1-p$. Both cases cause some overhead. The average overhead (in symbols) per frame equals

$$p\cdot\frac{1}{2}\frac{e_is_i}{s_j}\cdot\left(\frac{T}{e_is_i}-\frac{T}{e_js_j}\right)+ \qquad (2)$$
$$(1-p)\cdot\frac{1}{2}\left(e_j-\frac{e_is_i}{s_j}\right)\cdot\frac{T}{e_js_j}=\frac{T}{2}\left(1-\frac{e_is_i}{e_js_j}\right)\frac{1}{s_j},$$

where the first term represents the average overhead due to the switch (where q is uniform in $(0, e_is_i/s_j)$) and the second one due to the padding (where q is uniform in $(e_is_i/s_j, e_j)$).

B. Optimal scheduling orders

In this section we will consider all possible scheduling orders and proof the optimality of two specific orders: the $L \rightarrow H$ order (class $1, 2, \ldots, n$) and the $H \rightarrow L$ order $(n, n-1, \ldots, 1)$ and this for a wide variety of system setups.

More specifically, we show that if the MOD does not decrease as the MODCOD increases, $L \rightarrow H$ realizes the lowest possible overhead, while if the MOD does not increase as the MODCOD increases, $H \rightarrow L$ becomes optimal. Thus, when the MOD remains fixed, both orders generate the same mean overhead and are both optimal.

Let $e_{(1)}, e_{(2)}, \ldots, e_{(n)}$ be an arbitrary permutation of $e_1, e_2, \ldots, e_n$ that determines the scheduling order. Then, one readily obtains the following theorem from equations (1) and (2):

Theorem 1. *Let E_S be the scheduling overhead of the order S determined by $e_{(1)}, e_{(2)}, \ldots, e_{(n)}$. Then, provided that enough IP packets for each MODCOD are present at the start of the scheduling batch we have that E_S equals*

$$\frac{T}{2}\sum_{i=1}^{n}\left(1-\left(\frac{e_{(i)}s_{(i)}}{e_{(i+1)}s_{(i+1)}}\right)^{sgn(e_{(i+1)}s_{(i+1)}-e_{(i)}s_{(i)})}\right)\frac{1}{s_{(i)}},$$

where $s_{(n+1)} = s_{(1)}$, $e_{(n+1)} = e_{(1)}$ and $sgn(x)$ returns the sign of x.

For example, for the $L \rightarrow H$ (class $1, 2, \ldots, n$) scheduling order, this becomes

$$E_{LH} = \frac{T}{2}\left[\sum_{i=1}^{n-1}\left(1 - \frac{e_i s_i}{e_{i+1}s_{i+1}}\right)\frac{1}{s_i} + \left(1 - \frac{e_1 s_1}{e_n s_n}\right)\frac{1}{s_n}\right],$$

where the first $n-1$ terms correspond with all $L \rightarrow H$ $i \rightarrow i+1$ transitions between the different classes, and the last term with the $n \rightarrow 1$ $H \rightarrow L$ transition across batches.

Define $a_i = e_i s_i$ and $w_i = 1/s_i$, then an optimal scheduling order is one that maximizes:

$$f(S) = \sum_{i=1}^{n}\left(\frac{a_{(i)}}{a_{(i+1)}}\right)^{sgn(a_{(i+1)} - a_{(i)})} w_i.$$

Notice, any cyclic permutation of an order S results in the same amount of overhead. Therefore, when considering the overhead of an order, we can represent any order as a cycle with a clockwise orientation. Moreover, if the modulations are identical for all MODCODs, reversing the order does not alter the overhead.

Theorem 2. *The function*

$$f((x_1, y_1), \ldots, (x_n, y_n)) = \sum_{i=1}^{n}\left(\frac{x_i}{x_{i+1}}\right)^{sgn(x_{i+1} - x_i)} y_i,$$

where $x_{n+1} = x_1$, attains a maximum in

1) $(a_1, w_1), \ldots, (a_n, w_n)$ *if* $0 < a_1 < \ldots < a_n$ *and* $w_1 \geq \ldots \geq w_n \geq 0$,
2) $(a_n, w_n), \ldots, (a_1, w_1)$ *if* $0 < a_1 < \ldots < a_n$ *and* $0 \leq w_1 \leq \ldots \leq w_n$

among all permutations of $(a_1, w_1), \ldots, (a_n, w_n)$.

Proof: For $n = 2$ the result is trivial. We prove the first statement; the second can be derived in a similar manner. Define $(a_1, w_1), \ldots, (a_n, w_n)$ as the permutation L (or a cyclic permutation thereof) and let L' be an arbitrary permutation. Consider the cyclic representation of L and L'. For L, element (a_2, w_2) is positioned between (a_1, w_1) and (a_3, w_3), whereas for L' it is positioned between (a_k, w_k) and (a_j, w_j) for some $k, j \neq 2$ (see Figure 2).

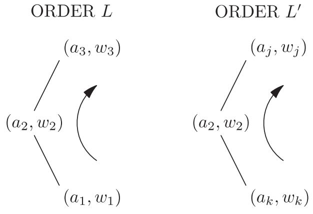

Fig. 2. Orders L and L'

Let L_r and L'_r be the same orders as L and L' after removing (a_2, w_2) from both orders, respectively. Then,

$$\begin{aligned} f(L) &= f(L_r) - \frac{a_1}{a_3}w_1 + \frac{a_1}{a_2}w_1 + \frac{a_2}{a_3}w_2, \\ f(L') &= f(L'_r) - \left(\frac{a_k}{a_j}\right)^{sgn(a_j - a_k)} w_k \\ &\quad + \left(\frac{a_2}{a_k}\right)^{sgn(a_k - a_2)} w_k + \left(\frac{a_2}{a_j}\right)^{sgn(a_j - a_2)} w_2. \end{aligned}$$

By induction, we know $f(L_r) \geq f(L'_r)$; hence, it suffices to show that $f(L) - f(L_r) \geq f(L') - f(L'_r)$. In doing so, we distinguish between four cases.

Case 1: $a_k > a_j > a_2$. The condition $f(L) - f(L_r) \geq f(L') - f(L'_r)$ can be rewritten as

$$\frac{a_j}{a_k}w_k + \frac{a_1}{a_2}w_1 + \frac{a_2}{a_3}w_2 \geq \frac{a_2}{a_k}w_k + \frac{a_1}{a_3}w_1 + \frac{a_2}{a_j}w_2,$$

where the i-th term on the left dominates the i-th term on the right, for $i = 1, 2$ and 3, which proves the statement.

Case 2: $a_j > a_k > a_2$. For this case we find that the condition $f(L) - f(L_r) \geq f(L') - f(L'_r)$ is equivalent to

$$\frac{a_k}{a_j}w_k + \frac{a_1}{a_2}w_1 + \frac{a_2}{a_3}w_2 \geq \frac{a_2}{a_k}w_k + \frac{a_1}{a_3}w_1 + \frac{a_2}{a_j}w_2.$$

The second term on the left still dominates the second one on the right, while the sum of the first and third on the left obeys the following inequality:

$$\frac{a_k}{a_j}w_k + \frac{a_2}{a_3}w_2 \geq \frac{a_2}{a_j}w_k + \frac{a_2}{a_k}w_2.$$

Therefore, it suffices to show

$$\frac{a_2}{a_j}w_k + \frac{a_2}{a_k}w_2 \geq \frac{a_2}{a_k}w_k + \frac{a_2}{a_j}w_2.$$

This statement is equivalent to having $(w_2 - w_k)(1/a_k - 1/a_j) \geq 0$, which holds in this particular case.

Case 3: $a_j = a_1$. In this case we need to prove the following expression:

$$\frac{a_1}{a_k}w_k + \frac{a_1}{a_2}w_1 + \frac{a_2}{a_3}w_2 \geq \frac{a_2}{a_k}w_k + \frac{a_1}{a_3}w_1 + \frac{a_1}{a_2}w_2.$$

Let us rewrite this expression as

$$a_1\left(\frac{w_k}{a_k} + \frac{w_1}{a_2} - \frac{w_1}{a_3} - \frac{w_2}{a_2}\right) \geq a_2\left(\frac{w_k}{a_k} - \frac{w_2}{a_3}\right). \quad (3)$$

As $(w_1 - w_2)(1/a_2 - 1/a_3) \geq 0$, the sum of the last three terms between brackets on the left dominates the last term between brackets on the right, thus

$$\begin{aligned} a_1\left(\frac{w_k}{a_k} + \frac{w_1}{a_2} - \frac{w_1}{a_3} - \frac{w_2}{a_2}\right) &\geq a_1\left(\frac{w_k}{a_k} - \frac{w_2}{a_3}\right) \\ &\geq a_2\left(\frac{w_k}{a_k} - \frac{w_2}{a_3}\right), \end{aligned}$$

where the last inequality follows from the fact that $a_1 < a_2$ and $w_k/a_k - w_2/a_3 \leq 0$. Remark, for $n = 3$, case 3 applies if $L \neq L'$.

Case 4: $a_k = a_1$. In this final case $f(L) - f(L_r) \geq f(L') - f(L'_r)$ does not necessarily hold, meaning we need another type of argument to complete the proof. For $a_j = a_3$, we have $f(L) - f(L_r) = f(L') - f(L'_r)$; hence, we may assume that $a_j \neq a_3$ as well. Instead of looking at the reduced permutations L_r and L'_r, define L_s and L'_s as the permutations obtained by removing a_3. Clearly, (a_3, w_3) is surrounded by (a_2, w_2) and (a_4, w_4) for L_s and by some elements (a_i, w_i) and (a_l, w_l) for L'_s. If $\min(a_i, a_l) > a_3$ we end up in a situation equivalent to case 1 or 2. As a_1 is followed by a_2 and a_2 is in its turn followed by $a_j \neq a_3$, the only possibility that remains is having $a_l = a_1$ and $a_i > a_3$. Rewriting $f(L) - f(L_s) \geq f(L') - f(L'_s)$ leads us to

$$a_1\left(\frac{w_i}{a_i}-\frac{w_3}{a_3}\right)+a_2\left(\frac{w_2}{a_3}-\frac{w_2}{a_4}\right)+a_3\left(\frac{w_3}{a_4}-\frac{w_i}{a_i}\right)\geq 0.$$

Replacing a_2 in this inequality by a_1 causes the left-hand side to decrease (as $w_2/a_3 - w_2/a_4 > 0$) and results in a statement equivalent to Eqn. (3). ■

The following observations can be made with respect to these theorems. When a transition to a higher MODCOD occurs, we should fill the remainder of the lower MODCOD frame by high MODCOD packets. A transition to a lower MODCOD frame either uses padding or switches the entire contents of the last higher MODCOD frame to a lower MODCOD, depending on which of the two strategies requires the least symbols. Under rather general conditions, the following conclusions can be drawn:

1) The *low to high* order causes the *least* overhead among *all* scheduling orders and this for any system setup in which a *higher* MOD implies a *higher* MODCOD (which will generally be the case in practice, see Figure 1).
2) The *high to low* order causes the *least* overhead among *all* scheduling orders and this for any system setup in which a *lower* MOD implies a *higher* MODCOD (which is highly unlikely in practice).
3) If the MOD can both decrease and increase while the MODCOD increases, neither the low to high or high to low order is necessarily optimal (which might occur within the DVB-S2 context).
4) If all the classes rely on the *same* MOD, the high to low and low to high order coincide (in terms of overhead) and are both optimal in achieving a minimal overhead.
5) The overhead for any system with at most 3 MODCODs, either coincides with the high to low or low to high order.

	Scen. 1	Scen. 2	Scen. 3	Scen. 4	Scen. 5
1	Q 1/4	8 1/4	16 1/4	8 2/3	8 1/2
2	8 2/5	8 2/5	8 2/5	8 3/4	Q 9/10
3	16 3/4	8 3/4	Q 3/4	16 2/3	8 2/3
4	16 9/10	8 9/10	Q 9/10	8 9/10	16 3/4

TABLE II

THE DIFFERENT SIMULATION SCENARIOS, EACH WITH FOUR QUEUES, SET UP WITH THE CORRESPONDING MODCOD FROM THE TABLE, SORTED ON ASCENDING SPECTRAL EFFICIENCY. (Q = QPSK, 8 = 8PSK, 16 = 16APSK)

C. Result validation via simulations

In order to validate the theorems, a simulation scenario has been set up, similar to Figure 3. This validation is based on comparing the simulated symbol overhead, with the theoretical results.

We model constant IP traffic, with a typical IP packet length distribution. This traffic has a number of destinations; each queue collects traffic for destinations with the same maximal MODCOD, and receives about the same amount of traffic. The outgoing link is set at a speed of 30 Mbaud/s, fully utilized by the incoming traffic. This causes the scheduler, which generates frames at link speed, to visit the same queue for a number of frames. Once started serving a queue, newly arriving packets are not taken into account until the next traversal. That is, the round robin system behaves like a polling system with gated access [6]. The time-driven simulation gathers information over 10000 seconds.

A total of 5 different system setups is considered (see Table II). Notice, in these simulation scenarios we do not necessarily restrict ourselves to the MODCOD combinations of the DVB-S2 standard, as our results are equally valid for more general cases. In a first series of simulations, we consider setups where the MOD either increases or decreases with the MODCOD. By calculating the number of symbols overhead per scheduling period (like E_S), we can thus validate Theorem 1. We start with a situation, similar to common DVB-S2 setups [7], where the MOD increases with the MODCOD. As shown by Theorem 1 and supported by the simulation (see Table III, scenario 1), $L \to H$ realizes the lowest overhead (in symbols). In the second scenario, where all MODs are equal, we see that both $L \to H$ and $H \to L$ are optimal. In the third scenario, where the MOD decreases when the MODCOD increases, being an unrealistic situation in DVB-S2, the simulations also confirm Theorem 1, with $H \to L$ being optimal.

For systems where a higher MOD does not imply a higher (lower) MODCOD, neither the $L \to H$ or $H \to L$ is necessarily optimal, as is shown in scenario's 4 and 5. In scenario 4, a combination of four MODCODs with one modulation out of order, shows that the optimal scheduling order is $1 \to 2 \to 4 \to 3$. However, the MOD does not need to decrease/increase with the MODCOD for the $L \to H$ or $H \to L$ to be optimal, as shown in scenario 5, where we have four MODCODs with one MOD out of order.

The simulation not only confirms the optimality result as stated by Theorem 2, but also reveals a close match with the number of overhead symbols as expressed by Theorem 1,

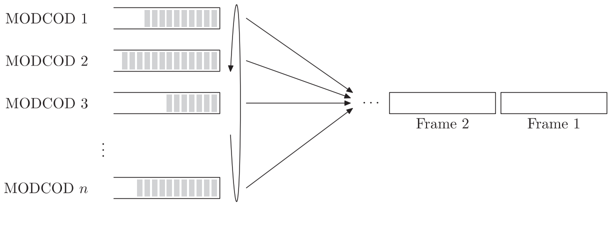

Fig. 3. Illustration of a DVB-S2 round robin scheduler.

	Scenario 1		Scenario 2		Scenario 3		Scenario 4		Scenario 5	
	Sim.	Theor.	Sim.	Theor.	Sim.	Theor.	Sim.	Theor.	Sim.	Theor.
$L \rightarrow H$	**24318**	**24323**	**18796**	**18785**	13565	13581	5810	5812	**11084**	**11088**
$1 \mid 2 \mid 4 \mid 3$	24809	24822	19126	19150	13219	13227	**5190**	**5188**	13687	13713
$1 \mid 3 \mid 2 \mid 4$	32582	32597	26136	26143	16958	16953	8591	8595	14321	14304
$1 \mid 3 \mid 4 \mid 2$	26604	26618	19154	19150	12743	12763	5820	5826	12293	12298
$1 \mid 4 \mid 2 \mid 3$	32638	32642	26174	26143	16770	16776	8348	8345	12949	12958
$H \rightarrow L$	26488	26525	**18778**	**18785**	**11479**	**11493**	5420	5427	11901	11925

TABLE III

SIMULATION AND THEORETICAL RESULTS. FOR THE FIRST THREE SCENARIOS, EITHER $L \rightarrow H$ OR $H \rightarrow L$ IS OPTIMAL, DEPENDING ON THE MOD ORDER. FOR THE LAST TWO SCENARIOS, A HIGHER MOD DOES NOT ALWAYS IMPLY A HIGHER (LOWER) MODCOD. IN SCENARIO 4 WE FIND THAT $1 \rightarrow 2 \rightarrow 4 \rightarrow 3$ IS OPTIMAL; WHEREAS IN THE LAST ONE $L \rightarrow H$ IS STILL OPTIMAL.

demonstrating the practical relevance of the model and overhead formula obtained. Also, even if the MOD can decrease and increase with the MODCOD, Theorem 1 quickly allows us to determine the optimal scheduling order numerically for n fairly small (say $n < 10$).

IV. CONCLUSION

We have presented a new theoretical model to analyze the padding and switching overhead for a series of packet scheduling orders. By constructing an analytical expression to determine this overhead, we were able to proof the optimality of some of the more natural orders, given some mild and realistic assumptions on the set of MODCODs in use. The various simulation runs performed did confirm the validity of both the model and the optimality result.

REFERENCES

[1] *Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for 11/21 GHz satellite services*, EN 300 421, European Telecommunications Standards Institute, 1997.

[2] A. Morello and U. Reimers, "DVB-S2, the second generation standard for satellite broadcasting and unicasting," *International Journal of Satellite Communications and Networking*, vol. 22, no. 3, pp. 249 – 268, 2004.

[3] O. Lucke, A. Jahn, and M. Werner, "MAC and encapsulation efficiency of satellite DVB using fade mitigation techniques," *International Journal of Satellite Communications and Networking*, vol. 24, no. 6, pp. 521 – 559, 2006.

[4] J. Cantillo, J. Lacan, and S. Combes, "A Design Rationale for Providing IP Services Over DVB-S2 Links," draft-cantillo-ipdvb-s2encaps-04, Internet Engineering Task Force, Dec. 2006. [Online]. Available: tools.ietf.org/html/draft-cantillo-ipdvb-s2encaps-04

[5] E. Çinlar, *Introduction to stochastic processes*. Prentice Hall, New York, 1975.

[6] H. Takagi, *Analysis of Polling Systems*. Cambridge Massachusetts: MIT Press, 1984.

[7] D. Breynaert and M. d'Oreye de Lantremange, "Analysis of the bandwidth efficiency of DVB-S2 in a typical data distribution network," March 2005, CCBN2005.