

Rule-based Packet Scheduling for DVB-S2 Through Generic Stream Encapsulation

Jean-Baptiste DUPÉ¹, Emmanuel CHAPUT², Cedric BAUDOUIN³, Caroline BÈS⁴, Arnaud DERAMECOURT⁴,
and André-Luc BEYLOT²

¹jean-baptiste.dupe@enseeiht.fr, Université de Toulouse; INP; TESA; F-31017 Toulouse, France

²{chaput,beylot}@enseeiht.fr, Université de Toulouse; INP; IRIT; F-31017 Toulouse, France

³cedric.baudoine@thalesaleniaspace.com, Thales Alenia Space

⁴{caroline.bes,arnaud.deramecourt},@cnes.fr, CNES

Abstract—In this paper, we consider the problem of IP packet scheduling over a GSE/DVB-S2 satellite link. Scheduling flows with QoS requirements has been widely addressed in the mobile field, especially in LTE and WiMAX, with emphasis on fairness, efficiency and dynamic adaptation to transmission conditions. We focus on the well-known empirical scheduling rules known as PF, M-LWDF and EXP-PF for both QoS and MODCOD scheduling, and present how they were adapted to GSE/DVB-S2 encapsulation. Some of the challenging issues yielded by DVB-S2 are tackled, such as joint scheduling of both QoS and MODCODs, concatenation of numerous user packets into one BBFrame and fairness issues introduced by the scheduling algorithm, especially when dealing with various transmission scenarios. We show the potential of our scheduling algorithm using several simulations.

I. INTRODUCTION

The rapid evolution of satellite use in recent years has led to the adoption of DVB-S2 [1], [2] and Generic Stream Encapsulation (GSE, [3]) standards, to reach high data rates, enable better Quality of Service (QoS) support and switching to all-IP architecture. Those evolutions gave rise to a growing need for better scheduling, that could provide guarantees for QoS-sensitive flows and support various traffic profiles.

Among the particular features of GSE, handling variable-length packets, through fragmentation and ACM (Adaptive Coding and Modulation) techniques are the most interesting yet challenging to consider. While introducing the necessary flexibility to achieve high data rates and low delay, they bring a complexity which is hard to take into account in the scheduling process. Our goal here is to build a joint scheduler, who could encompass information from several layers, such as QoS requirements, link quality and system load into a single scheduling process.

Previous work on this topic [4], [5], [6], [7], has been focusing on various aspects of DVB-S2 scheduling, such as cross-layering [6], [7], fairness [4] and GSE scheduling [5]. This paper tries to go further by adapting successful scheduling rules defined in the terrestrial wireless networks to satellite communications. This approach, while being more practical than classical optimization techniques [5], [6] seems to be a promising trade-off between performance and simplicity. It uses classical results drawn from utility theory combined with classic scheduling paradigms.

The remainder of this paper is organized as follows. In Section II we present the satellite system studied, and what difficulties arise from this model. Section III presents the different scheduling rules, and discuss their adaptation to the context. Section IV features the implemented scheduling algorithms, along with simulation results and analysis. Before concluding in section VI, we focus on fairness issues in section V.

II. SYSTEM DESCRIPTION

A. DVB-S2 and GSE

We consider a gateway serving several terminal clients through a geostationary satellite, using DVB-S2 and GSE on the forward link and a classic return link, such as DVB-RCS. Packets received at the gateway are encapsulated following the GSE and DVB-S2 protocols, as featured in figure 1.

Packets coming from higher layers are first encapsulated into GSE EPU, and possibly fragmented, before being sent into Base Band Frames (BBFrames). BBFrames used in DVB-S2 can be very large, to guarantee a quasi error-free transmission, thus several GSE EPUs can be concatenated into a single BBFrame.

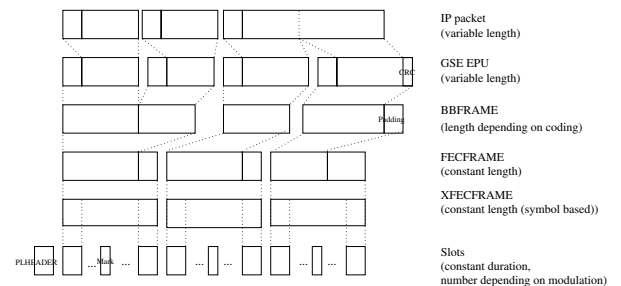


Fig. 1. DVB-S2/GSE encapsulation

The use of ACM supposes that the transmission adapts to the receiver's transmission conditions, in order to make a better use of the available bandwidth. Each user reports periodically its channel quality, which gives the modulation and coding (MODCOD) suited for this user, and therefore the associated BBFrame it has to be served with. Using different modulations and coding for BBFrames supposes that their payload and

transmission time depends on their MODCOD, making the scheduling decisions more complex. We call a MODCOD better than another one if the spectral efficiency of the first one is higher. Users associated to higher MODCODs can be served with a lower MODCOD in case there is still room in a BBFrame of lower MODCOD about to be sent. This technique is called reclassification.

Lastly, concatenation of several packets into one BBFrame introduces the problem of BBFrame filling ratio : even if fragmentation is used, it introduces overhead, as specified in GSE standard, and might lead to performance decrease if not properly used. In some cases, especially when the room left in the BBFrame is tiny, it might be more efficient to send the BBFrame without fragmenting, instead of introducing overhead. The BBFrame filling ratio is thereby an important parameter.

For the sake of clarity, we will only consider IP packets, albeit GSE can handle a wide set of protocols, and is not limited to IP. Since IP packets have variable size and can be fragmented, we will make full use of the specific features of GSE.

B. Joint BBFrames and GSE packets scheduling

Scheduling of GSE over DVB-S2 introduces specific issues, mainly due to the concatenation of several packets into a variable payload, and the variable transmission time. Classic scheduling techniques, such as Round-Robin or Fair Queuing only consider one user¹ at a time, but scheduling of GSE packets yields two overlapping decisions: choosing which BBFrame to transmit, and then choosing which users to serve in it. The choice of the BBFrame has deep consequences: users with worse MODCOD cannot be served in this BBFrame, and users with better MODCOD should not be served in priority, in order to maximise spectral efficiency (avoid large reclassification).

Classical approaches on DVB-S2 scheduling usually consider two-steps based schedulers [6], [7], based on WRR and Proportional Fairness techniques. The first step is dedicated to user scheduling, that is which and in what order IP packets will be served, and the second step consists on choosing which BBFrame is to be transmitted next. While this approach shows good results and low scheduling complexity, it is usually difficult to parameter, and has limited flexibility, mainly due to the two distinctive scheduling steps. Recent work on GSE scheduling [4] shows how simple metrics can be used to schedule both packets and BBFrames, but focuses on a single metric at a time, while our goal is to build a joint scheduler, taking advantage of several parameters from lower and higher layers. The latest approach in [5] uses utility theory to schedule BBFrames through a knapsack algorithm, showing key challenges of GSE scheduling, and how fragmentation and reclassification can be used to improve performances. It however yields a high computational cost.

In this paper, we consider *scheduling rules* as a mean to encompass several parameters into one decision. They consist on

a function, defined for each user, evaluated at every scheduling decision. This function is defined empirically, but strives to follow the same behaviour as utility functions, making use of classical results (such as Proportional Fairness), yet altered to fit to the needs of the scheduling objective (delay, fairness, loss rate, etc.).

This type of scheduling has been widely used in wireless networks [8], [9] but never in our context. Satellite communications show anyhow distinctive characteristics, especially on the channel attenuation profiles and time scale, but also on the number of users: while a traditional base station has a few dozen active users at a time, the satellite system we aim to study contains several hundred users. Furthermore, scheduling rules are defined for user-by-user scheduling, and our system needs to actually schedule BBFrames, hence the need for a different design. Bearing in mind those specifics is fundamental to design efficient rules for DVB-S2 scheduling.

III. SCHEDULING RULES

Scheduling rules known as PF, EXP-PF and M-LWDF, have been proposed in [10], for use in HDR/CDMA networks, and are now considered as classic for LTE networks. They make use of a scheduler with N users (or queues), and serve one queue, hence one packet, at a time. To adapt them to our specific context, we need to schedule BBFrames, containing several packets. The adapted rules must therefore indicate the next MODCOD to schedule. Table I features the different notations we will use.

N	Total number of active users
N_k	Total number of users using MODCOD k
$N_k^a(n)$	Number of active users on MODCOD k an instant n
$r_i(t)$	Instantaneous achievable rate for user i
$\bar{r}_i(t)$	Mean rate for user i
$W_i(t)$	Head of line delay for user i
R_k	Instantaneous rate of a BBFrame with MODCOD k
$\tau(n)$	Transmission time of BBFrame n
$L_{i,k}(n)$	Number of bits served for user i, MODCOD k
$c_{i,k}$	User i, MODCOD k rule
$\gamma_{i,k}$	Coefficient of user i, MODCOD k
$a_{i,k}$	QoS parameter, user i MODCOD k
j	Selected user
m	Selected modcod

TABLE I
NOTATIONS

Note that, for our scheduler, each variable has two indexes: one for the user, and the second for its associated MODCOD. For example, $r_{i,k}(t)$ is the instantaneous achievable rate for user i , MODCOD k . Moreover, time can be seen as a succession of BBFrames, we thus note the time corresponding to the n th BBFrame $t(n)$ or simply n .

A. Proposed Adapting

To adapt scheduling rules to our joint user and MODCOD (BBFrame) scheduling, we define two rules: the user rule and MODCOD rule. *User rule* is defined for each user, taking into account various parameters relevant to this particular user, such as rate, delay, loss rate. *MODCOD rule* is a function of the user rules belonging to this MODCOD, thus encompassing user

¹user, flow or queue are defined as an entrant to the scheduling algorithm and used without distinction in this paper.

needs, but also adds parameters belonging to this MODCOD: payload and transmission time. Selection of IP packets within a BBFrame is done through user rules, and MODCOD selection is done via MODCOD rules. The continuous usage of the same functions allows us to provide a coherent scheduling scheme, from IP packets to BBFrames. We will first show the adaptation process with PF, then with M-LWDF and EXP-PF.

PF (for Proportional Fairness), introduced by Kelly in [11], has been extensively covered [12], [13], [14], especially concerning wireless networks, and is considered as a reference in scheduling. It can be derived from utility theory [15], where it is proven to maximize the total utility of the system, respective to throughput. Its expression can be derived as follows:

$$j = \arg \max_i \frac{r_i(t)}{\bar{r}_i(t)}$$

This rule offers a trade-off between high achievable throughput and fairness: it tends to favour flows with higher instantaneous achievable rate, provided all users have the same mean rate. This behaviour is called *opportunistic scheduling*. Dividing by the mean rate balances the opportunistic scheduling by also favouring flows with low mean rates. PF usually applies to *elastic traffic*, with low QoS requirements, such as best effort.

The first step in our adaptation is the achievable rate. Wireless link capacity is embodied in the MODCOD choice and a better MODCOD will have a higher achievable rate. It thus seems reasonable to replace $r_i(t)$ with R_k : the scheduler will favour better MODCODs. However, this parameter is common to all users, and has no influence on user scheduling. Hence, we define the user rule simply as:

$$c_{i,k}(n) = \frac{1}{\bar{r}_{i,k}(n)}$$

The MODCOD rule must then take into account the achievable rate, but also the mean rate. Instead of taking the mean MODCOD rate in the MODCOD rule, we simply sum over each active user, to reflect the offered load, and balanced by the active user ratio. The idea is to give virtually "worse" channel condition to MODCODs with few active users, who might not have enough traffic to fill the BBFrame.

$$m = \arg \max_k R_k \frac{N_k^a(n)}{N_k} \sum_{i=1}^{N_k} c_{i,k}$$

A noticeable issue is the throughput measure. As stated in [5], classical exponential moving average formulas fail to give a good estimation of throughput when the transmission time, thus the measurement period, is not constant. Hence, the average rate for user i , MODCOD k is measured with the following formula:

$$\bar{r}_{i,k}(n) = \alpha^{\tau(n)} \bar{r}_{i,k}(n-1) + (1 - \alpha^{\tau(n)}) \frac{L_{i,k}(n)}{\tau(n)}$$

where α is close to one.

B. Modified Latest Weighted Deadline First

Modified Latest Weighted Deadline First (M-LWDF) uses the head of line delay as the main parameter to schedule

packets. This rule is well adapted to real-time traffic and has also been used for mixed traffic types. It is throughput-optimal, in the sense that the queues remain stable, if any other rule can achieve stability [10]. It is given by:

$$j = \arg \max_i \gamma_i r_i(t) W_i(t)$$

where γ_i is a parameter embodying the QoS requirements.

Following the same approach as PF, we define the updated M-LWDF modcod rule as:

$$m = \arg \max_k R_k \frac{N_k^a(n)}{N_k} \sum_{i=1}^{N_k} \gamma_{i,k} W_{i,k}(n)$$

Where the user rule is simply:

$$c_{i,k}(n) = \gamma_{i,k}(n) W_{i,k}(n)$$

The QoS parameter, $\gamma_{i,k}(n)$ is set to:

$$\gamma_{i,k}(n) = \frac{a_{i,k}}{\bar{r}_{i,k}(n)}$$

This allows to use the classical PF ratio, but balanced by the delay. Parameter $a_{i,k}$ is usually defined following QoS requirements, such as:

$$P[W_{i,k}(n)] > W_{max,i,k} = \delta_{i,k}$$

The maximum delay, $W_{max,i,k}$ is set according to the type of traffic (VoIP, Streaming or Best Effort). [10] recommends the following value of a_i :

$$a_{i,k} = \frac{-\log(\delta_{i,k})}{W_{max,i,k}}$$

This adaptation is not straightforward: since multiple packets of the same queue can be served, taking only the head of line delay $W_{i,k}$ could lead to inefficient solutions, related to the way we fill up BBFrames after choosing the MODCOD to send. We however assume that the delay differences are relatively small, particularly under high load, and have limited impact on the applications.

C. Exponential Rule

Exponential rule (EXP-PF) [10] tries to smooth the delay differences by giving exponential priority to users with higher delay. Based on PF, it is also an opportunistic scheduler. By using exponential priority, the scheduling rule overrides the channel benefits when the delays become too large. It is equivalent to PF when all users have the same delays. EXP-PF is given by:

$$j = \arg \max_i \gamma_i r_i(t) \exp \left(\frac{a_i W_i(t) - \overline{aW}}{1 + \sqrt{aW}} \right)$$

where $\overline{aW} = \frac{1}{N} \sum_{i=1}^N a_i W_i(t)$.

Similarly to PF and M-LWDF, we propose the following adaptation to our context:

$$m = \arg \max_k R_k \frac{N_k^a(n)}{N_k} \sum_{i=1}^{N_k} \gamma_{i,k} \exp \left(\frac{a_{i,k} W_{i,k}(n) - \overline{aW}}{1 + \sqrt{aW}} \right)$$

Note that the mean weighted delay, \overline{aW} is common to all users, not just users within a MODCOD.

D. Rule-based Scheduling algorithm

Our proposed scheduling algorithm is done in two steps: rule calculation and BBFrame allocation, linked by the use of the same scheduling rules. In the first step, each user and MODCOD rule is calculated, and the results gives the next MODCOD (BBFrame) to be scheduled. Then, the second step consists on the allocation of the BBFrame payload, following the user rules. The allocation algorithm is based on a quick sorting of users according to their rule value. The scheduler takes all the packets from the user with the best MODCOD and user rules and places them into the BBFrame. If room is still available, it repeats the same operation (sorting, selecting and filling) with the second best, and so on until either there is no room left in the BBFrame or users have no packets left. In the latter case, we can use reclassified packets of users with a better MODCOD, whom packets are chosen with the same procedure.

IV. SIMULATION

A. Simulation parameters

The specificity of our system had to be taken into account to parameter the simulations, by considering which particular features were of importance regarding the behaviour of the system. First, the large number of users (eg. several thousands) on a single system gives narrow margin concerning the implementation of the algorithm: a flow-based scheduling is hard to consider at this level, while user-based (eg. terminal) scheduling brings few advantages. We have chosen a DiffServ approach, where user packets are sent into queues following their PHB. Our system includes three PHB, known as EF, AF and BE, with decreasing QoS requirements. Four different MODCODs are considered, representing classical "Clear Sky" scenarios, where most users have the best MODCOD, and very few the worse one. The MODCOD characteristics are displayed on table II. Simulation duration is of 100 seconds, to suppress the influence of the transient period, and the input load is 0.9.

k	MODCOD	BBFrame payload (bits)	Rate
0	QPSK 1/3	21408	6.61 Mbps
1	8-PSK 2/3	43040	20 Mbps
2	16-APSK 3/4	48408	30 Mbps
3	32-APSK 5/6	53840	41.5 Mbps

TABLE II
MODCODS USED

To analyse the behaviour of our system, we focused on two parameters: the mean delay and the reclassification ratio. The delay is of utmost importance for QoS flows, and the reclassification ratio gives an idea of the spectral efficiency of the algorithm. In addition, we have monitored the BBFrame filling ratio, which might show inefficient behaviours, and is related to the system load.

B. Proportional Fairness

Proportional Fairness (figure 2), is clearly unstable in our scenario, the best effort queue of the best MODCOD, the most

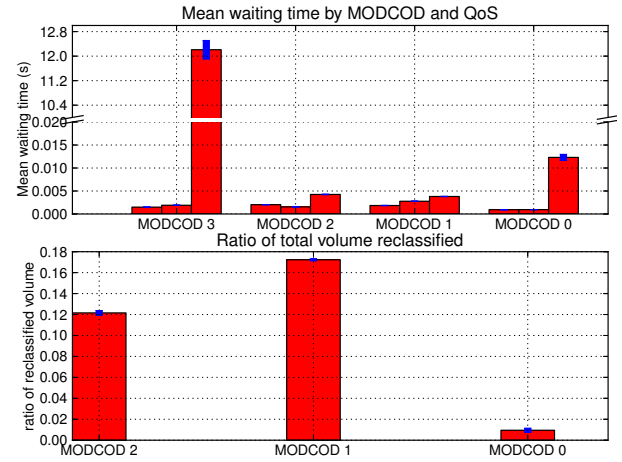


Fig. 2. Proportional Fairness

heavily loaded one, having an ever growing delay (please note the scale gap). This result is not only showing that our version of proportional fairness is not well adapted to the system, but clearly shows the limits of PF itself : in case of very uneven load repartition, and under high load, delays are potentially very large, even though the rates measured are proportionally fair. Other queues show nonetheless delays under 15 ms, far below the QoS requirements for Best Effort traffic (200 ms). It also shows poor reclassification volume. This is due to the fact that even if a user is reclassified, its MODCOD was not chosen at this scheduling round. In this situation, the user rates are updated, because they have been granted some room in the BBFrame, and thereby influences the choice of the next MODCOD, making users who have been reclassified less likely to have their MODCOD selected. This phenomenon disturbs the balance of the system, who see its performance dwindle as lower MODCODs are more often selected, even if they have less data to be transmitted. However, removing the reclassification makes the overall system capacity theoretically lower, and will not improve performance. No large disparities of filling ratio were observed, the mean was found to be 0.95, higher than the offered load due to the GSE overhead.

C. M-LWDF rule

Figure 3 shows large improvements compared to PF, both in terms of delay (all of them are below 70 ms), and in terms of spectral efficiency, the relative volume of reclassified data being low, around 10%. The most obvious reason for this large performance improvement is that the delay balances the behaviour of Proportional Fairness: MODCODs under high load will quickly see their delay rising if they are not served, thus giving them priority, and allowing the system to be much more reactive than with PF. The second reason is more related to the scenario we have chosen: our scheduling rule selects the last two MODCODs at most. More than 80% of the total throughput originates from those MODCODs, corresponding loosely to the offered load (but not only), hence the delays are kept without difficulty below the QoS requirements. The reclassification volume is significant only for the MODCOD 1.

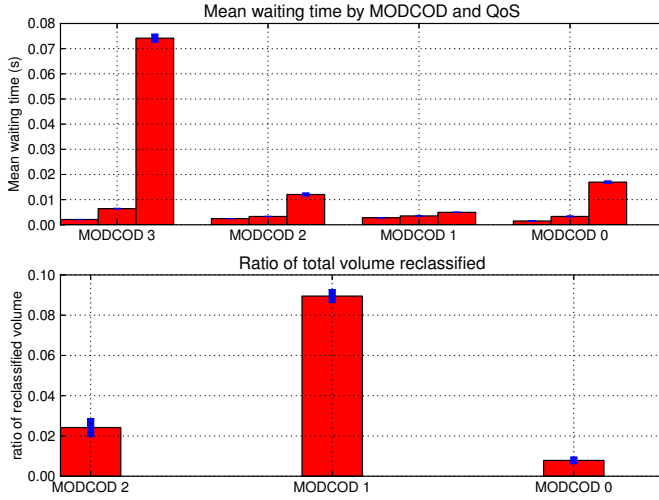


Fig. 3. M-LWDF Rule

The reason is to be found again in the opportunistic behaviour: compared to MODCOD 0, MODCOD 1 has a better spectral efficiency, and is thus scheduled more often. However, its offered load is sometimes not enough to use the full link capacity, and room is still available for MODCOD 2, triggering reclassification. The better MODCODs have enough load to fill their BBframes up. Here again, the filling ratio was the same for all MODCODs, with a value of 0.95.

D. EXP-PF rule

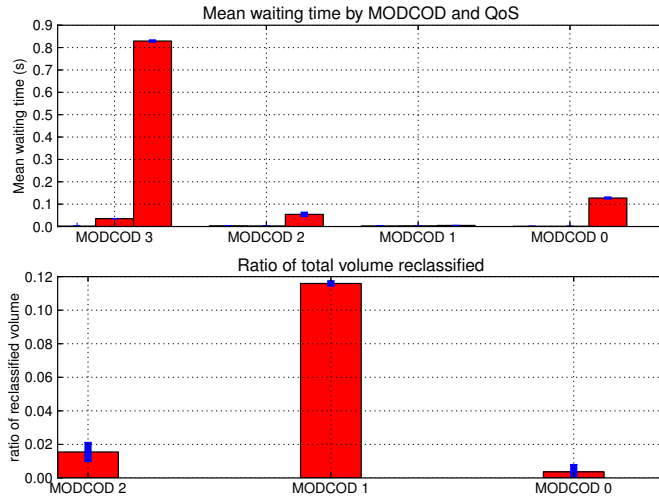


Fig. 4. EXP-PF Rule

EXP-PF rule performs somewhat as a trade-off between PF and M-LWDF. Delays are bounded, and below QoS requirements for almost all flows, except the Best Effort in MODCOD 3. Since EXP-PF is "fairer" than M-LWDF, the best MODCODs are not selected as often, and, under our particular load distribution, this leads to a relative inefficiency in terms of delay and mean spectral efficiency. EXP-PF gracefully adapts to delay differences in the single-user mode, but our proposed

rule leads to lower performance for high load flows, because its behaviour approaches PF, inefficient in terms of delay.

V. FAIRNESS

A. Controlling Fairness

Fairness is a key notion often broached in research on resource allocation and scheduling, yet it is hard to define properly in mathematical terms. One of the most widely recognized measures of fairness is Jain's Fairness Index [16]. It measures quadratic ratio of user's rates, thereby assuming all users have to receive the same mean rate, which might not always be the case. Jain's Fairness index is defined as:

$$\mathcal{J} = \frac{(\sum_{i=1}^N r_i)^2}{N \cdot \sum_{i=1}^N r_i^2}$$

Many papers, using utility functions, introduce a Fairness Parameter [6], [17], presented as a way to control the algorithm's Fairness. In our case, this fairness parameter (β) is used to control the influence of the rate in the scheduling decisions. For example the PF user rule is now:

$$c_{i,k} = \frac{1}{(\bar{r}_{i,k}(n))^\beta}$$

This parameter allows to give more or less priority to users with higher rates, henceforth enforcing some kind of trade off between performance and fairness. It can be useful in our case: while fairness among users seems to be quite understandable, fairness among MODCODs is a different issue. Having this parameter allows us to control fairness, unlike fixed criterion algorithm, such as the maximization of spectral efficiency: favouring users with higher rate can improve it, while still being fair.

B. Simulations

Our previous simulations show how being "Fair" between users depends on how those users are defined, and on what type of scenario we are in. As we considered users with very different input loads and QoS requirements, applying fairness can lead to inefficiencies. Figure 5 shows how the mean delay and fairness index evolve with β (delay in log scale). We took M-LWDF as a reference algorithm, and β ranging from 0.1 to 5. Jain's fairness index is calculated for each MODCOD.

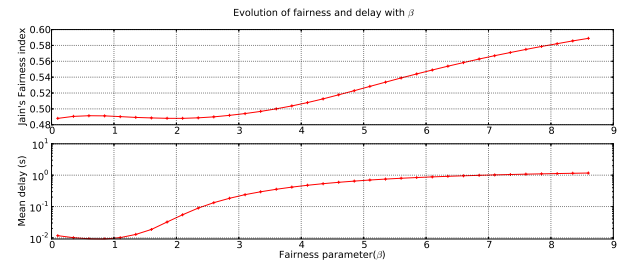


Fig. 5. M-LWDF Delay and fairness

As expected, fairness improves with β , as the throughputs of each MODCODs become fairer, but the mean delay increases,

as lower MODCODs are more often served, reducing the system's capacity and resulting in a delay increase. It should be noted that the delay and fairness are only affected by values of β above 1.4, which shows that fairness can be tuned in a limited range without lowering performances : even if overall fairness does not change, MODCOD throughputs are affected. It also seems that values between 0.5 and 1 give the lowest delay, it corresponds to the where fairness is best adapted to the system load, considering delay performance.

Those results show how the fairness parameter can be adapted to give the best behaviour, and on what range it can be tuned without affecting global performances. In the "Clear Sky" scenario, load disparities between the MODCODs require an "unfair" value of β to give the best performance. To assess this behaviour, we take a scenario where each MODCOD is loaded evenly, with equivalent users in terms of QoS. The results are displayed on figure 6.

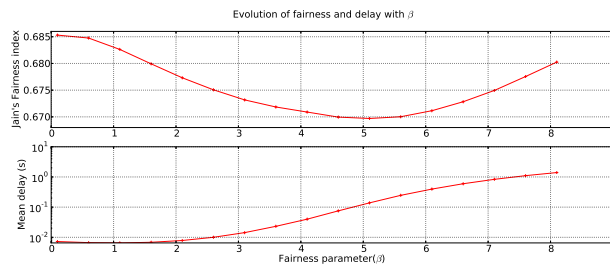


Fig. 6. M-LWDF Delay and fairness - uniform load

In the uniform case, the β value giving the best mean delay is situated between 1 and 1.4, supposedly fairer values, which corresponds to the scenario. Interestingly, the fairness index is decreasing on the chosen range, although the magnitude is low compared to the previous scenario. The reason is to be found in the combined influence of rate and reclassification: the inflexion point of the curve is where the decrease in rate for better MODCODs is not balanced by the reclassification in worse MODCODs.

VI. CONCLUSION

Scheduling rules, inherited from utility theory, are an efficient way for joint scheduling, taking into account parameters from several layers and different QoS requirements. We have adapted those rules to GSE/DVB-S2 and studied their performance in terms of introduced delay and rate, which - to our knowledge - has not been done before in this context.

The defined rules show promising results, and the adaptation done allows to take into account some of the specific features of GSE/DVB-S2, especially concatenation and variable-length payload, due to ACM techniques. The rules hold a low computation cost, provided the number of users in the system is limited, which is the case if we consider a DiffServ paradigm. We have also shown that tuning fairness can further improve performance.

Yet many challenges remain to correctly adapt scheduling rules to GSE/DVB-S2. First and foremost, the reclassification and BBFrame filling ratio are key performance indicators, and

are not yet included into the rules, we plan to study this influence. The filling algorithm is also of great importance, and we only have considered a basic algorithm. Improving it, by taking into account other parameters, could be the next step of our study. Also, since scheduling rules were proven to be stable, it should be possible to do the same with our defined rules, although the context is more complex to model. Lastly, the behaviour of the rules under a degradation of transmission conditions could be interesting to focus on.

REFERENCES

- [1] EBU-UER, "Digital video broadcasting (DVB) ; second generation framing structure, channel coding and modulation systems for broadcasting, interactive services, news gathering and other broadband satellite applications," Tech. Rep. v1.1.2, European Telecommunications Standards Institute, June 2006.
- [2] EBU-UER, "Digital video broadcasting (DVB) user guidelines for the second generation system for broadcasting, interactive services, news gathering and other broadband satellite applications (DVB-S2)," Tech. Rep. v1.1.1, European Telecommunications Standards Institute, February 2005.
- [3] DVB-GS. W. Group, "IP/S.2 study, GSE protocol specification," Tech. Rep. Document rev 10, European Telecommunications Standards Institute, 2007.
- [4] E. Chaput, A.-L. Beylot, and C. Baudoin, "Packet Scheduling Over DVB-S2 Through GSE Encapsulation," *IEEE GLOBECOM 2008 - 2008 IEEE Global Telecommunications Conference*, pp. 1–5, 2008.
- [5] E. Chaput, M. Verloop, A.-L. Beylot, and C. Baudoin, "Utility function based packet scheduling over DVB - S2," *2013 IEEE International Conference on Communications (ICC)*, 2013.
- [6] D. Pradas and M. a. Vazquez-Castro, "NUM-Based Fair Rate-Delay Balancing for Layered Video Multicasting over Adaptive Satellite Networks," *IEEE Journal on Selected Areas in Communications*, vol. 29, pp. 969–978, May 2011.
- [7] M. A. V. Castro and F. Vieira, "DVB-S2 Full Cross-Layer Design for QoS Provision," *IEEE Communications Magazine*, no. January, pp. 128–135, 2012.
- [8] S. Shakkottai and A. L. Stolyar, "Scheduling for Multiple Flows Sharing a Time-Varying Channel : The Exponential Rule," *Translations of the American Mathematical Society-Series 2*, vol. 207, pp. 185–202, 2002.
- [9] B. Sadiq, R. Madan, and A. Sampath, "Downlink Scheduling for Multiclass Traffic in LTE," *EURASIP Journal on Wireless Communications and Networking*, vol. 2009, no. 1, p. 510617, 2009.
- [10] S. Shakkottai and A. L. Stolyar, "Scheduling algorithms for a mixture of real-time and non-real-time data in HDR," in *Proceedings of 17th International Teletraffic Congress*, pp. 793–804, 2000.
- [11] F. Kelly, "Charging and rate control for elastic traffic," *European Transactions on Telecommunications*, vol. 8, no. 1, pp. 33–37, 1997.
- [12] G. Song, Y. Li, and L. J. Cimini, "Joint channel- and queue-aware scheduling for multiuser diversity in wireless OFDMA networks," *IEEE Transactions on Communications*, vol. 57, no. 7, pp. 2109–2121, 2009.
- [13] M. Andrews, Q. Lijun, and A. Stolyar, "Optimal Utility Based Multi-User Throughput Allocation subject to Throughput Constraints," *24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2005)*, vol. 4, pp. 2415–2424, 2005.
- [14] P. Hosein, "QoS control for WCDMA high speed packet data," *4th International Workshop on Mobile and Wireless Communications Network*, no. 4, pp. 169–173, 2002.
- [15] G. Song and Y. Li, "Utility-based resource allocation and scheduling in OFDM-based wireless broadband networks," *IEEE Communications Magazine*, vol. 43, no. 12, pp. 127–134, 2005.
- [16] R. Jain, D.-M. Chiu, and W. R. Hawe, *A quantitative measure of fairness and discrimination for resource allocation in shared computer system*. Eastern Research Laboratory, Digital Equipment Corporation, 1984.
- [17] V. Kavitha, E. Altman, R. El-Azouzi, and R. Sundaresan, "Fair Scheduling in Cellular Systems in the Presence of Noncooperative Mobiles," *2010 Proceedings IEEE INFOCOM*, vol. 0, pp. 1–9, Mar. 2010.