OAM Technique Based on ULE for DVB-S/S2 Transmission Systems

Raffaello Secchi, Nimbe Ewald, and Gorry Fairhurst

Electronics Research Group (ERG),
School of Engineering, Aberdeen King's college
Fraser Noble Building, AB24 3UE, Aberdeen, UK
{raffaello,nimbe,gorry}@erg.abdn.ac.uk

Abstract. This paper presents a technique to monitor the performance of DVB-S/S2 transmission systems at the Link Layer (LL) level using the Unidirectional Lightweight Encapsulation (ULE) protocol. The technique is based on the use of extension headers and is validated for different traffic patterns. It has been designed to provide an Operation, Administration and Management (OAM) method for IP-based DVB-S/S2 systems using the MPEG-2 Transport Stream. Such methods can be used to identify edge-to-edge performance and connectivity issues in a layer 2 network.

Keywords: DVB-S, DVB-S2, ULE, OAM.

1 Introduction

Digital Video Broadcast for Satellite (DVB-S/S2) [1] standards have been used to successfully deploy standard transmission networks with over a hundred million DVB-S receivers deployed worldwide. DVB-S/S2 not only provides a platform that may be used for traditional television broadcast services, but may also be used to support a growing number of IP-based and interactive services [2,3,4]. The growing use of IP services, both for TV broadcast, augmented services, backhaul/contribution and Internet access is about to revolutionise the satellite service portfolio.

In general, Internet Service Providers (ISPs) and content providers do not manage a DVB-S/S2 satellite platform directly. They typically lease satellite access and bandwidth resources from satellite service providers or third-party operators. This hierarchy is common to many other transmission networks – ISPs often utilise the service offered by transmission operators. Therefore, an ISP that needs to manage the Quality of Service (QoS) guarantees provided to a customer must be able to separately validate the quality of the underlying transmission service.

One method to verify the operation of a service offering is to make performance or connectivity measurements across a portion of the transmission network, sometimes called edge-to-edge measurements between the ingress and egress of a

G. Giambene and C. Sacchi (Eds.): PSATS 2011, LNICST 71, pp. 104–115, 2011.

transmission network segment (such as DVB-S/S2 transmission system). These measurements must necessarily be below the IP or Layer2 Virtual Network Provider (L2VNP - RFC 4664), because the end-to-end network services (e.g. including allocation of IP addresses) are not the responsibility of the transmission operator.

Methods of this form provide an important tool for Operations, Administration, and Management (OAM) in most managed terrestrial networks. Edge-to-edge measurements can help in verifying performance of the transmission service and also are particularly valuable in the location of faults, by indicating whether a reported issue is a result of a transmission fault, network fault, configuration error, etc.

In many cases, current methods are based on proprietary tools that pertain to specific network architectures [5,6]. However, standardisation groups are also realising the importance of having common tools, although the individual methods need to be related to the specifics of the transmission technology (e.g., OAM for MPLS [7], the Metro Ethernet Forum [8] and the Telecommunication Standardisation Sector (ITU-T) [9], IEEE 802.3ah [10] and 802.1ag [11] standards for Ethernet). However, no work has been published to specify OAM methods for DVB-based satellite system, specifically at the link layer.

The common techniques for OAM used on DVB networks rely on monitoring signal level and synchronisation at the physical later and on IP-level techniques for bi-directional services such as netperf or ICMP (ping). DVB systems allow also Transport Stream metrics to be monitored, such as the Continuity Counter, or the Adaptation-Field timestamp. However, these are not easily related to the packet performance experienced by individual edge-to-edge paths through the transmission network.

This paper proposes an OAM technique based on the Ultra-Lightweight Encapsulation (ULE) method. The ULE protocol stack provides a multi-protocol encapsulation for a MPEG-2 TS that supports an extension-header mechanism that can be used to tag any packet sent on a transmission link with additional information – in this case a timestamp option [13] that can be utilised by an OAM framework. We define the use of this timestamp and explain how it may be utilised to monitor the QoS performance of DVB-S/S2 endpoints at the link-layer (e.g., delay, jitter, ordering, and bandwidth consumption). This method is validated using a set of traffic profiles.

The remainder of this paper is as follows: a brief description of ULE and its features is given in Section 2. Section 3 describes the performance monitoring techniques used in this work, while Section 4 describes the testbed used and shows the experimental results. Finally, Sections 5 and 6 provide the discussion, conclusions and future work.

2 Ultra Lightweight Encapsulation

DVB-S/S2 systems adapt variable-sized MPEG-2 Audio/Video payloads to fixed-size 188 B Transport Stream (TS) packets. This may be used to transmit IP datagrams using adaptation protocols such as Multi-Protocol Encapsulation (MPE) [1] or Unidirectional Lightweight Encapsulation (ULE) [12]. MPE specifies a

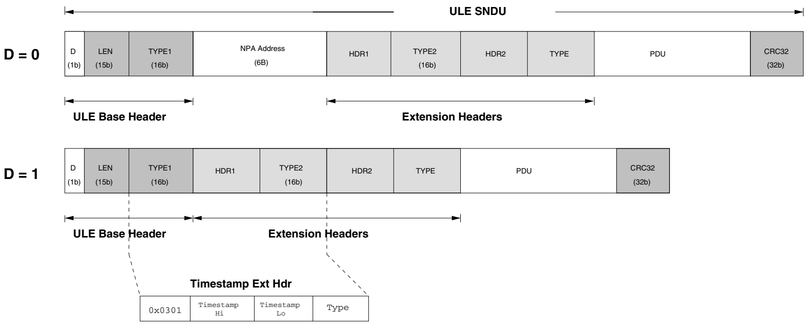

Fig. 1. Format of the service network data unit of ULE

method to adapt variable-size network data packets (typically IP packets) to fixed-size 188 B TS packets. It builds upon the section syntax of the MPEG standards and is widely deployed for IP encapsulation over DVB-S/S2 systems. Its minimum header is 12 B (plus a cyclic redundancy check field – CRC-32) while the maximum payload size is typically 1.5-4 kB (1 kB in some systems, e.g., [14]). Many systems restrict use of MPE to IPv4 traffic, although in principle it could be extended to other payloads using a LLC/SNAP extension or by defining new interpretations of the payload.

ULE adopts an IP-centric encapsulation. It mandates only the functions strictly necessary for transmission over a transport stream (the base header has only three fields header in contrast to eighteen of MPE). This improves encapsulation performance requiring a minimum base header only 4 B long, but also supporting a maximum payload of 32 kB. ULE was defined to be multi-protocol, so it can readily support IPv6, MPLS, IEEE 802.1pQ VLAN tags, etc. ULE also supports an extension header mechanism similar to IEEE 802, and IPv6 that allows optional additional information to be added to a encapsulated packet when needed. This also allows new extensions to be added to the protocol when new applications and needs emerge (e.g., the timestamp option introduced in RFC 5163). This contributes in improving network/receiver-processing efficiency. Several research papers have been published on encapsulation efficiency achieved by MPE and ULE [15,16,17,18].

2.1 ULE Header

The ULE header is shown in Fig.1. The first three fields form the base header, these are: Destination Address Absent flag (D), Length and Type, form the 4 B base header. If the D-bit is 0, the 6 B link-layer receiver address is present in the header at the end of the base header. The 15-bit Length field indicates the size in bytes of the ULE packet from the byte following the base field to the last byte of the CRC-32. The 16-bit Type field indicates the type of network packet using IEEE ether-type convention (e.g., 0x0800 for IPv4 or 0x86DD for IPv6) carried in the ULE payload. Type fields lower than 1536 (decimal) correspond to extension headers.

More specifically, a ULE extension header is identified by a 16-bit Type field, which specifies a header length field (H-LEN) field and a header type field (H-Type) [13]. Optional extension headers are identified by H-LEN values from one to five, while the value zero is reserved for mandatory extension headers. The H-LEN indicates the length in bytes for each optional extension header. Receivers not required to process the extension header can therefore access the next header, skipping as many bytes as indicated in H-LEN.

The receiver is required to extract and process every mandatory extension header present in the ULE packets (and to silently discard payloads with an unknown type). In contrast, a receiver can silently ignore optional extension headers without impairing communication. In other words, the expression optional in the context of extension headers refers to the receivers ability to process or not the extension header, not to the transmitters requirement to insert it. The payload associated is always passed to the higher layer protocol handling, irrespective of whether the option is understood. An optional extension header can therefore be used to carry OAM signalling, which may be utilised by specific terminals.

2.2 The Timestamp Extension Header

RFC 5163 [13] defines three types of ULE extension headers, MPEG-2 TS-Concat, PDU-Concat and Timestamp. The first two are mandatory extension headers designed to carry more than one (TS or IP) packet in a single ULE payload. The Timestamp header is an optional extension header aimed to support monitoring and measurement of an operational ULE link.

The first byte of the base header (TYPE1) is 0x03, indicating a total length of 6 B. The extension header then comprises two fields: the timestamp (divided into an HI and a LO part), and the next type field. The 32-bit timestamp records the number of microseconds past the hour in Universal Time Coordinates (UTC) recording the time of encapsulation. A timing resolution of one microsecond is sufficient for OAM and performance evaluation.

The Timestamp extension header used in our experiments complies with RFC 5163 and is shown in Figure 1. Since we encapsulated IPv4 packets and we did not include any layer-2 address (D=1) or extension headers apart from the timestamp, the ULE overhead per IP packet was 14 B (RFC 4326 identifies cases where this format may be safely used). This is a rather small overhead with respect to the maximum transmission unit (MTU) of Internet packets. Moreover, as shown in the results section, not every packet needs to carry a timestamp to achieve good OAM performance. Accurate link monitoring can be achieved by inserting a timestamp every several tens of kilobytes, significantly reducing the timestamp overhead cost in terms of capacity.

3 Traffic Monitoring

Systems for traffic monitoring are developed to assess performance, detect faults, etc. The level of required accuracy depends on the application. The assumptions here are that the service will be monitored at the receiver. The collected data may be retrieved either by a return channel or collected by some other means, though this is not the subject of the current paper.

This paper considers two performance indexes commonly used for traffic monitoring [3]: one-way delay (OWD) and delay jitter. The timestamps for the i-th packet collected at sender and receiver are respectively indicated by s_i and r_i. The OWD and the delay jitter are defined as:

$$\begin{aligned} owd_i &= r_i - s_i \\ jitter_i &= |owd_{i+1} - owd_i| \end{aligned} \tag{1}$$

These metrics may be used for all packets using a TS or only packets associated with specific flows (e.g., identified by a particular DiffServ code-point when supporting multiple levels of QoS). If ULE encapsulation is performed as soon as packets are delivered to the satellite interface, the OWD takes into account any queueing delays at the terminal. This allows network operators to verify SLAs for delay (either for all traffic or specific high-priority flows). Also, measurements of sender and receiver bit rates can help identify the causes of delay, namely users not respecting the SLA maximum allowed bit rates or ISPs under-provisioning network capacity.

Synchronisation between sender and receiver is necessary to measure the one-way delay. The receiver needs to obtain a reference clock, e.g. using the Global Positioning System (GPS), the Network Time Protocol (NTP), or by reconstructing the clock present in the TS network clock reference (NCR). The reference clock is not required to measure delay jitter, since this only requires compensating the drift between sender and receiver clocks. In both cases, the two endpoints need to agree on the method to insert the timestamps to reconstruct sender bit rate estimations at the receiver.

We implemented our proposed method using a timestamp attached to a ULE-encapsulated packet every time a predetermined amount of data has been sent (partially including the packet that carries the timestamp). This method (byte counting) is considered more accurate [17] than inserting a periodic timestamp every N packets when packets length are subjected to jitter.

Several factors influence the accuracy of measurement. First, since the transmitter and receiver may run several real-time processes at the same time, the timestamp could not necessarily reflect the exact transmission or reception time of a packet whether a process pre-empt the receiving/transmitting routine. This problem is platform-dependent, which is typically offset by employing dedicated hardware.

If ULE timestamps may be used to perform bandwidth measurements, the ULE timestamps may not take into account MPEG overhead. Two methods, packing and padding, can be used to encapsulate ULE packets in MPEG TS packets [16]:

– In padding mode, each packet is encapsulated and put into a TS packet independently from other packets in a flow. If the last fragment of a ULE

packet does not fit the TS payload, padding bytes are added to complete the 188 B packet.
- In packing mode, data from the next ULE packet can be used to complete a TS payload. Thus, a ULE packet in packing mode does not have to start at the beginning of a TS packet. If the encapsulator is not able to complete a TS packet because no data is available, it starts a timer. If new data is available within a pre-determined time interval, the encapsulator completes and delivers the packet; otherwise, when a timer expires, the encapsulator releases the packet with padding of the remaining bytes.

In addition to increasing encapsulation efficiency by reducing the average amount of padding per cell, packing mode also increases the estimation accuracy when using ULE timestamps.

Packet loss can be a cause of inaccurate estimations. A receiver error may induce erroneous bandwidth estimation if the ULE packet carrying the timestamp is lost. In many circumstances lost packets can be detected and not included in the statistics even without the presence of a sequence number. For instance, if the sender inserts a timestamp every N packets, the receiver is able to identify a missing timestamp when more than N packets have been received and no timestamps have been detected. In general, the probability to acknowledge an incorrect timestamp sample decreases as N increases. As observed in the next section, collecting a timestamp every 10-20 kB (or approximately every 10 packets) could be a good compromise between sampling rate and robustness of estimation against packet losses.

4 Experimental Results

This section describes our Linux-based testbed used to evaluate the suitability of ULE as an OAM tool. The testbed implemented an IP over DVB-S2 stack using ULE. The TS was received and analysed by the real-time MPEG decoder, dvbsnoop, which was adapted to support ULE.

4.1 Test-Bed Description

The testbed consists of a unidirectional DVB-S2 link between two Linux computers. The sender has a DVB-S2 modulator (Advanced Digital TVB590). An Application Program Interface (API) allowed transmission parameters (frequency, symbol rate and coding rate) to be controlled. The receiver comprises a DVB-S2 card (Hauppauge WinTV) [19] configured to process the transmitter TS. The transmission rate was fixed to 1 Mb/s. In some cases, the bandwidth was varied enforcing the insertion of blocks of null TS packets at regular intervals. This emulated a Constant Rate Allocation (CRA) in a Time Division Multiplexing (TDM) environment where a station receives a periodical transmission opportunity. The NCR was inserted periodically to ensure synchronisation between transmitter and receiver.

A user-space ULE/MPEG encapsulator was developed based on tcpdump packet capturing library. The program captures Ethernet frames and builds ULE/MPEG encapsulated packets in a circular buffer. A concurrent process manages (using a modulator API) the transfer of 4 kB-data blocks from the buffer to the DVB card buffer. The Ethernet Type field is copied into the ULE Type field. If the Type field is not present (the Ethernet Type/Length field is less than 1536), the ULE Type is described by a logical link control/sub-network access protocol (LLC/SNAP) header. In this case, the encapsulator operates in bridging mode [12] copying the entire Ethernet frame into the ULE payload field.

The encapsulator supports both padding and packing encapsulation modes. In our experiments, the packing mode was used to increase efficiency. The timeout duration (packing threshold) was set to 100 ms as recommended in [16] for multimedia traffic. The timestamp is the only extension header inserted in ULE packets.

Figure 2 shows two video traffic profiles from traces used in our tests. Both traces are carried over UDP with an average transmission rate of 64 kb/s. The flow using H.261 encoding carries a TV stream characterised by high bit rate variability (the peak-to-average ratio is about 4.7 over 200 ms). The H.264-encoded stream is a video-chat yahoo trace with more regular bit rate (the peak-to-average is 1.6 over 200 ms). These two different profiles allowed the evaluation of estimator properties for different level of traffic burstiness.

The real-time flows were multiplexed with other Internet traffic, including web browsing and file transfers. The competing traffic was captured on a LAN and replicated on the DVB link multiplexed with the video flow. The encapsulator

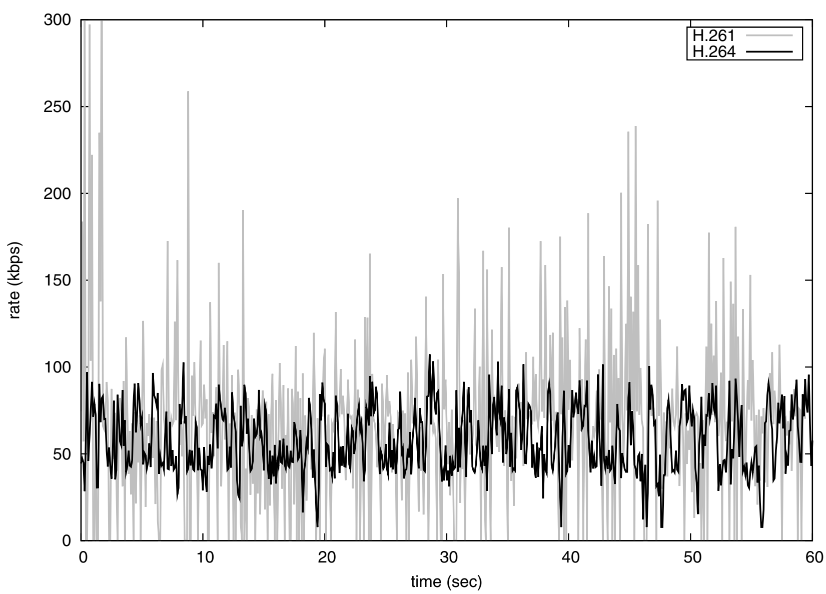

Fig. 2. Traffic profiles used in experiments

inserted the timestamps on video packets only. We tested two cases: 1) the competing traffic has a bit rate comparable to that of the video flow (low load) and 2) the competing traffic presents a bit rate approximately ten times larger (high load) than that of the video flow. Although the overall link capacity is not fully utilised in both scenarios, the competing traffic causes considerable jitter and queueing delays especially in high-load scenario. This is due to the high level of burstiness of HTTP traffic, which manly consists of relatively small and frequent data transfers.

4.2 Results

Figures 3 and 4 display the complement of the cumulative distribution function (CDF) of OWD estimates for the H.261 and H.264 traces, respectively. Each graph reports the CDF for the high and low-load scenarios when the fraction of video packets with timestamp (SR) is 5%, 10% and 100%, e.g., if SR=100%, all video packets contain a timestamps.

In the high-load scenario the competing traffic causes large delays to video packets of both traces. However, increasing the workload impacts differently the two types of traffic. For instance, the proportion of packets exceeding one second OWD is larger in H.264 traces (approximately 24% vs. 18%). On the other hand, the tail of CDF drops faster in H.261 graphs than H.264, which means that H.261 (TV) packets can experience large delays with higher probability.

This discrepancy is mainly due to the different level of burstiness of the two streams. The H.264 flow has a more regular pattern, H.264 packets tend to be more spread, resulting in a higher probability of meeting a burst of cross traffic and, consequently, higher OWD delays. The large rate variations of H.261 flows also result in longer delay if a queue is congested. The timestamp method is able to reveal the OWD behaviour with high accuracy in both scenarios, even when only a small fraction of packets (5%) contain a timestamp. This means that infrequent packet probes are sufficient to discover LL delay anomalies and monitor the queueing behaviour.

Figures 5 and 6 show the complement of jitter CDF for H.261 and H.264 traces, respectively. Each graph reports statistics from the high-load and low-load scenarios in top and bottom diagrams, respectively. Again, the CFD is displayed when 5%, 10% and 100% of packets are sampled. Since jitter measures the variation between consecutive OWD samples, major differences between jitter CDFs in the low-load and high-load scenario are observed when the timestamps are far apart. It is observed that if all the video packets are sampled, the majority of jitter samples lay between 0 and 200 ms and jitter distributions are alike. Conversely, taking a timestamp every 20 packets produces much more visible variations (spreading samples between 200 ms and 1200 ms when the queue is loaded), which allows detecting easily congestion conditions. As a result, a few jitter samples, which represent a low overhead, are sufficient to detect the queue state.

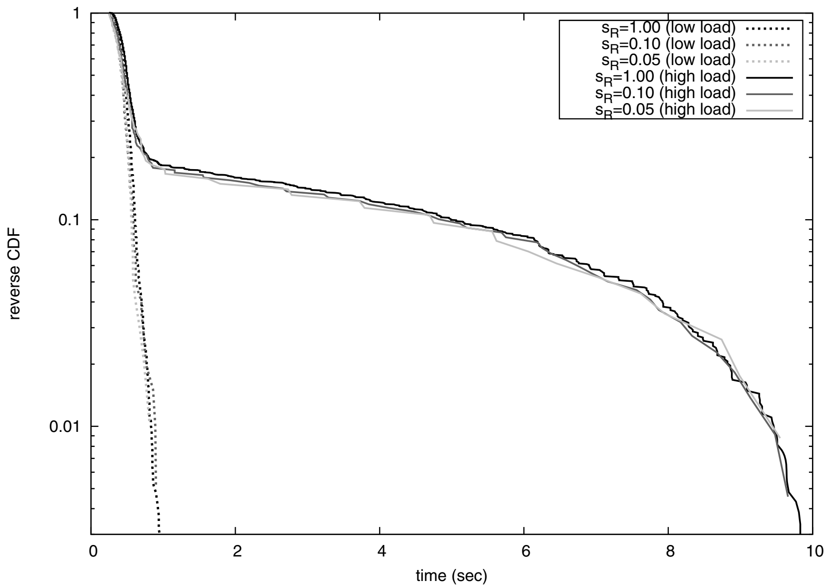

Fig. 3. Reverse CDF of OWD for the H.261 flow

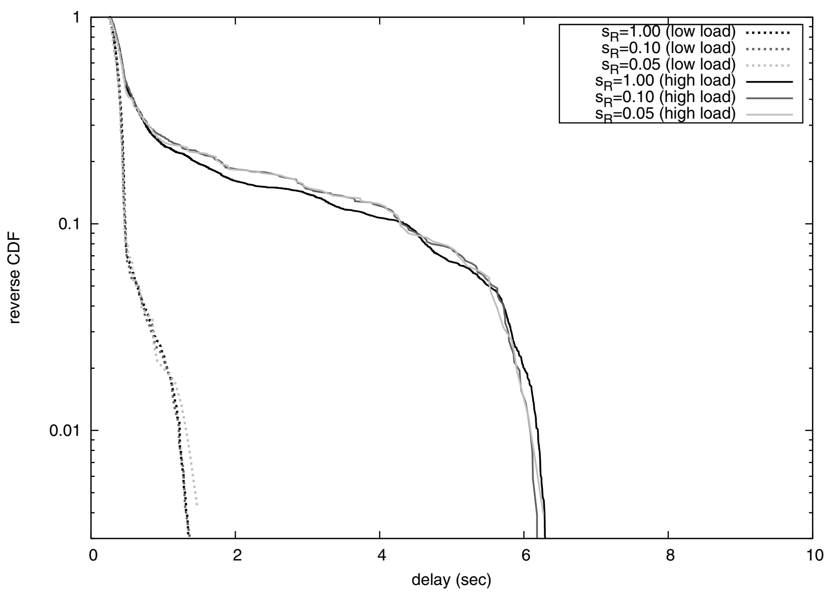

Fig. 4. Reverse CDF of OWD for the H.264 flow

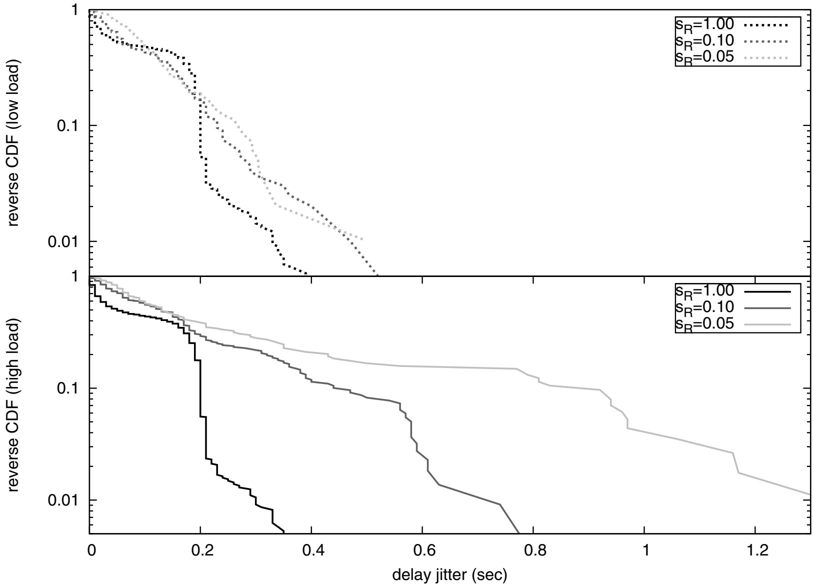

Fig. 5. Reverse CDF of jitter for H.261 traffic

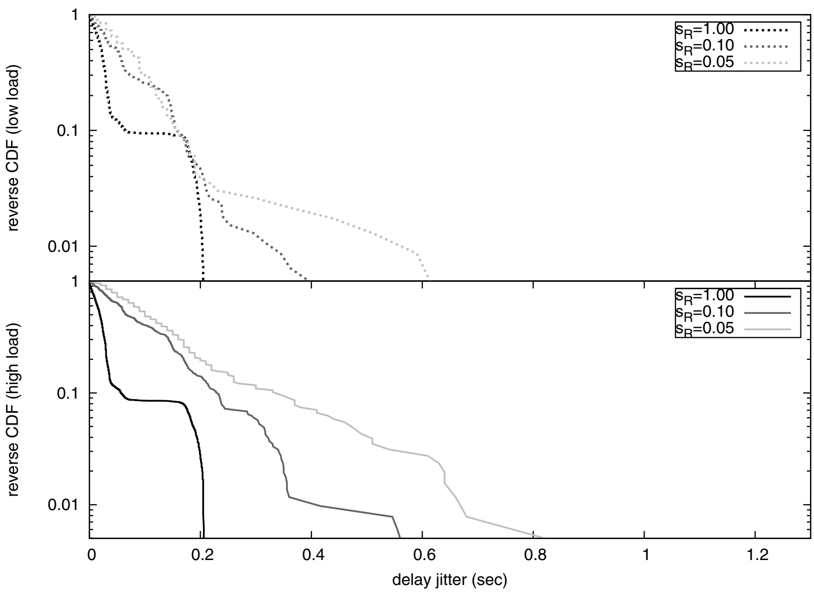

Fig. 6. Reverse CDF of jitter for H.264 traffic

5 Future Directions

The paper focuses on unidirectional streams, and performance measurement at the receiver. Further work may analyse the effectiveness of the method with a range of deployment scenarios and traffic types. In addition, a return link is desirable to transmit performance data to the sender.

When a return link is provided, the architecture could be extended to verify ordering and perform other test by comparison with a log of the sequence of packets transmitted at the sender. It may also be possible to use a mandatory header to carry an echo request formatted as a mandatory TEST extension (e.g., to verify a remote receiver has processed a specific packet, although not the focus of the present paper), allowing pro-active testing of the stream.

This paper describes use of an OAM method for ULE. In addition to ULE, DVB-S2 also supports the Generic Stream, using the GS Encapsulation (GSE) protocol. GSE supports direct transmission of IP traffic over the DVB-S2 baseband frame (i.e., TS packets are not used), but since ULE extension headers were developed to be compatible with GSE, a similar OAM timestamp method could be used with GSE. This method would need to be subtly different in the way timestamps were utilised, and this requires further work.

DVB transmission systems such as DVB-T and DVB-C could also be used with ULE and DVB-T2 and DVB-SH also support GSE. There is therefore potential to develop a common OAM technique that span a range of DVB deployment scenarios.

6 Conclusions and Future Work

This paper presents an implementation of ULE as an OAM tool to monitor DVB-S/S2 transmission systems. We show that the timestamp option can be used effectively for this purpose. This extension header provides timely OAM data with low overhead. It can be used to check compliance with a given service level agreement, to verify system performance and to assist in locating a performance or connectivity issue.

We developed a proof-of-concept testbed to implement this new tool. ULE timestamps allow several system parameters to be monitored such as user bandwidth consumption, one-way delay, delay jitter, and packet losses. In particular, the one-way delay (OWD) is a critical measure for assessing performance for real-time applications. The delay jitter of the timestamps can be used to detect increasing levels of congestion or non-compliance in the delay levels of a SLA. Our measurements confirm that good estimates can be achieved even for low packet sampling rates. However, OWD measurements need the synchronisation of the sending and receiving terminals, which may require separate synchronisation channels. Our experiments suggest that when properly tuning the sampling rate is possible to detect a range of network conditions. In general, a low sampling rate (e.g., one packet every 200 ms) is sufficient for this purpose.

References

1. EN 301 192, Digital Video Broadcasting (DVB); DVB specification for data broadcasting. ETSI Standard (2009)
2. TR 102 033, Digital Video Broadcasting (DVB); Architectural framework for the delivery of DVB-services over IP-based networks. ETSI technical report (2002)
3. TS 102 034, Digital Video Broadcasting (DVB); Transport of MPEG-2 TS Based DVB Services over IP Based Networks. ETSI standard (2008)
4. TS 102 826, Digital Video Broadcasting (DVB); Guidelines for the implementation of DVB-IP Phase 1 specifications. ETSI (2009)
5. Fujitsu white paper, Ethernet Service OAM: Overview, Applications, Deployment, and Issues (2010), `http://www.fujitsu.com/downloads/TEL/fnc/whitepapers/EthernetService-OAM.pdf`
6. Cisco systems white paper, Overview of Ethernet Operations, Administration, and Management (2010), `http://www.cisco.com/en/US/prod/collateral/routers/ps368/prod_white_paper0900aecd804a0266.html`
7. Yasukawa, S., Farrel, A., King, D., Nadeau, T.: RFC 4687, Operations and Management (OAM) Requirements for Point-to-Multipoint MPLS Networks. IETF informational standard (2006)
8. Technical Specification MEF 17, Service OAM Requirements and Framework Phase 1, Metro Ethernet Forum (MEF) (2007)
9. ITU-T rec. Y.1731, OAM functions and mechanisms for Ethernet based networks. ITU rec. (2006)
10. IEEE 802.3ah, Ethernet in the First Mile. IEEE standard (2004)
11. IEEE 802.1ag, Connectivity Fault Management. IEEE standard (2007)
12. Fairhurst, G., Collini-Nocker, B.: RFC 4326 Unidirectional Lightweight Encapsulation (ULE) for Transmission of IP Datagrams over an MPEG-2 Transport Stream (TS). IETF standard (2005)
13. Fairhurst, G., Collini-Nocker, B.: RFC 5163 Extension Formats for Unidirectional Lightweight Encapsulation (ULE) and the Generic Stream Encapsulation (GSE). IETF standard (2006)
14. ATSC Doc. A/94, ATSC Standard: ATSC Data Application Reference Model. Advanced Television Systems Committee (2002)
15. Montpetit, M.-J., Fairhurst, G., Clausen, H., Collini-Nocker, B., Linder, H.: RFC 4259 A Framework for Transmission of IP Datagrams over MPEG-2 Networks. IETF informational standard (2005)
16. Collini-Nocker, B., Fairhurst, G.: ULE versus MPE as an IP over DVB Encapsulation. In: HET-NETs 04 Second Intl. Working Conf., pp. Y70/l–P7019 (2004)
17. Fairhurst, G., Mathews, A.: Internet Protocols over Digital Video Broadcasting Media (IP over DVB) Study of Encapsulation and Protocol Performance (2003)
18. Cantillo, J., Collini-Nocker, B., De Bie, U., Del Rio, O., Fairhurst, G., Jahn, A., Rinaldo, R.: GSE: A flexible, yet efficient, encapsulation for IP over DVB-S2 continuous generic streams (2008)
19. WinTV-DVB-s WinTV-Nexus_s, Product Description (15/02/2010), `http://www.hauppauge.com/html/dvb_s.htm`