RTP4MUX: A NOVEL MPEG-4 RTP PAYLOAD FOR MULTICAST VIDEO COMMUNICATIONS OVER WIRELESS IP

Abdelhamid Nafaa, Toufik Ahmed, Yassine Hadjadj Aoul, Ahmed Mehaoua

CNRS-PRiSM Lab., University of Versailles
45 av. des Etats-Unis, 78035 – Versailles - France
Tel: +33 1 39 25 40 76
Fax: +33 1 39 25 40 57
{anaf, tad, yana, mea}@prism.uvsq.fr

ABSTRACT

The ISO/IEC MPEG-4 multimedia content creation, management and distribution framework is foreseen to be an important component of many forthcoming Wireless IP Multimedia services. However, current approaches for transporting MPEG-4 Elementary streams over IP networks are not optimized for wireless group communications. To address this issue we propose a novel Real-Time Transport Protocol (RTP) payload, called RTP4mux, that provides better data multiplexing and flow aggregation over shared wireless IP links. RTP4mux offers the following transport features (1) an MPEG-4 elementary streams interleaving mechanism that minimize temporal coding dependencies between adjacent RTP packets and then improving burst packet loss tolerance (2) a configurable two levels Access Units multiplexing scheme, that optimizes wireless bandwidth utilization and reduces end-to-end transmission delays through a lower data control overhead and a shorter packetization latency.

Keywords: Wireless IP, RTP payload, MPEG-4, QoS.

1. INTRODUCTION

Wireless IP, Wireless LAN and MPEG-4 technologies are rapidly becoming core components of next generation standard-based multimedia networks and services. Hence, the integration of voice/video/data communications and wireless networks is a topic of intense research interest [1], [2] and [3]. In the other hand, the development of application-layer protocols for the real-time distribution of audio and video over IP has been under study for several years [4], [5] and [6]. The most prominent protocol is the Real-Time Transport Protocol (RTP) described in [7]. RTP is a generic protocol that should be tailored according to media classes and exigencies. It is accompanied by documents (payload formats) that describe the specific encoding of different media types (RTP profile). RTP provides end-to-end network transport features suitable for real-time applications, such as audio and video over multicasts or unicast IP network. RTP is implemented on top of UDP/IP stack. It does not provide any multiplexing scheme. Transporting MPEG-4 Group communications media streams over Wireless Links using RTP experiences many problems among which traffic overhead, and transmission delays. The overhead increases when transporting each Elementary Stream (ES) as a separate RTP session. While the end-to-end transmission delays are in part introduced by the RTP packetization process.

MPEG-4, described in [8] and [9], is a recent ISO/IEC standard for the coding of natural and synthetic audio-visual data in the form of audiovisual objects (AVOs). Each AVO is represented as one or many Elementary Streams (ESs). AVOs are arranged to form an MPEG-4 scene by means of a scene description (SD). The MPEG-4 standard provides functionalities which go far beyond the scope of its well established predecessors, namely, MPEG-1 and MPEG-2. It includes features which will facilitate the transmission and the QoS management of media streams over a broad range of network protocols such as ATM and IP.

This article focuses on the design of an efficient multicast transport protocol for MPEG-4-based group communications over wireless IP networks. Scalability, Bandwidth optimization and Low latency are properties required for such a end-to-end real-time transport protocol. The proposed RTP4mux protocol is intended to be used on Wireless IP Gateways and Bridges to mix several RTP sessions over a shared wireless link. It defines a new RTP payload format for MPEG-4 audio/video elementary streams that offers (1) an appropriate ESs fragmentation and data interleaving to improve error robustness over noisy wireless links (2) an ESs multiplexing scheme, that reduces the end-to-end delays and the traffic overhead, (3) a many-to-one RTP session mixer for higher scalability.

The remainder of this paper is as follows. Section 2 introduces the MPEG-4 terminal architecture and outlines the key components for conveying MPEG-4 streams over the IP network. Section 3 reviews and compares the various approaches on transporting MPEG-4 video streams over RTP/UDP/IP. In Section 4, we present the proposed RTP4mux protocol. Section 5 is devoted to the performance evaluation and simulation results analysis. Finally, we conclude in Section 6.

2. MPEG-4 OVERVIEW

MPEG-4 is a significant expansion on the idea of MPEG-2, the previous video coding standard, which is used for applications such as DVD and digital television. In particular, the MPEG-4 standard provides the following enhancements compared to MPEG-2:

Media object representation: Rather than simply describing a video stream frame-by-frame, an MPEG-4 scene is described as a set of media objects. A media object may be a still image (such as a fixed background), a video object (such as a talking person), or an audio object (such as a voice corresponding to another media object). Media objects can be either 2 or 3-dimensional, and can be composed of either natural (i.e. pixel-based) or synthetic data. ESs are partitioned into AUs (Access Units). Access Units are defined by MPEG-4 general framework for the framing of semantic entities in Elementary Streams (Examples: A valid MPEG-4 Elementary Stream could be an MPEG-1 video, labeled with MPEG-4 system information in its headers. An AU would then be one video frame I, B or P). Those AUs will be labeled with priority information, timing information, and others.

Scene description: The hierarchical relations, locations, and properties of ESs in a video sequence are described by dynamic set of ODs (Object Descriptor). OD's are themselves conveyed through one or more dedicated ES's. MPEG-4 defines a binary language for composing a set of multiple media objects into a scene. This language is derived from VRML (Virtual Reality Modeling Language) [10] and known as BIFS (BInary Format for Scenes). Scene description is also conveyed in a separate elementary stream.

Multiplexing and synchronization of data streams: Synchronization is achieved by time stamping individual AUs within elementary streams. MPEG-4 defines a tool called "FlexMux" [11], which defines a method for grouping elementary streams. Multiplexing is used to group together elementary streams which have a similar quality of service (QoS) requirements in order to reduce the number of network connections. Multiplexing is achieved using the ES_ID (Elementary Stream Identifier).

2.1. Layered MPEG-4 System Architecture

The MPEG-4 terminal architecture is composed of three layers: Compression Layer, Synchronization Layer and Delivery Layer (see Figure 1).

The Compression Layer [11], [12], [13] organizes the ESs into Access Units (AU). The AU is the smallest data element that can be attributed an individual timestamps. The concept of an Access Unit defines the boundary between media specific processing and delivery specific processing. That means that the transport should not depend on the nature of the media data but only on AU properties.

The Sync Layer (SL) [11] provides the synchronization between streams and defines a homogeneous encapsulation of ESs carrying media or control data (ODs, BIFS). In this layer, the consecutive data from one stream is called an SL-packetized stream. The interface between the compression layer and the Sync Layer is called the Elementary Stream Interface (ESI). The notion of ESI is only informative i.e. it is extremely useful in order to define concepts and mechanisms but does not have to be implemented.

The Delivery Layer in MPEG-4 [14] consists of the Delivery Multimedia Integration Framework. It provides transparent access to media by abstracting the delivery technologies used. The interface between the SL and DMIF is called the DMIF Application Interface (DAI). It offers content location independent procedures for establishing MPEG-4 sessions and access to transport channels.

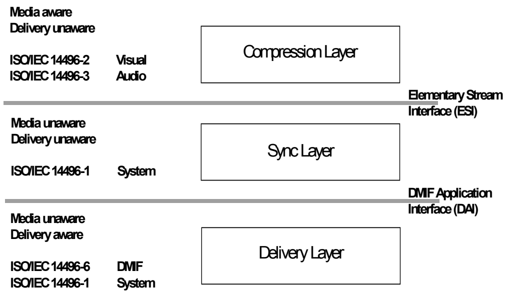


Figure 1: MPEG-4 System Architecture.

2.2. MPEG-4 ES transport over IP networks

At the sending side, the compression layer compresses the visual information and generates elementary streams (ESs), which contain the coded representation of AVO. The ESs are packetized as SL-packetized streams at the Sync Layer. The SL-packetized streams provide timing and synchronization information, as well as fragmentation and random access information. The SL-packetized streams are multiplexed into a FlexMux stream at the Delivery Layer [14], which is then passed to the transport protocol stacks such as RTP/UDP/IP. The resulting IP packets are transported over Internet.

At the receiver side, the video stream is processed in the reversed manner before its presentation. The received

AVOs are decoded and buffered before being composed by the player.

There are a number of RTP packetization schemes for MPEG-4 data. Many works presented in [15], [16], [17], [18], [19] and [20] specify how MPEG-4 streams are fragmented and packetized to be conveyed over IP network. Many packetization schemes can be implemented together in one terminal. Each packetization scheme is basically adapted to a particular media stream. This technique is called Media-aware packetization. For example a video frames is fragmented at a recoverable sub-frame boundaries to avoid error propagation.

It is likely that several RTP packetization schemes will be needed to suit the different kinds of encoding media types (CELP audio, video simple profile, …) and multimedia networking services (VoD, streaming Audio, wireless videoconferencing, …).

3. RELATED WORKS

RTP provides end-to-end network transport features much suitable for real-time applications, such as audio and video over multicast or unicast IP network services [7]. Some early researches have addressed the problem of transporting MPEG-4 streams over IP. We describe and compare these contributions in the following.

3.1. RTP Payload Format for MPEG-4 Audio/Visual Streams

This approach is defined in [15]. It remains actually the only IETF (Internet Engineering Task Force) standard for transporting the MPEG-4 Audio/Visual streams over IP networks. This specification seems to be the most suitable and realist for coordination with the existing Internet QoS. It uses the MPEG-4 Simple profile Level 1 coding with only one video object in the scene. Nevertheless, it does not use MPEG-4 systems at all. The Internet RFC 3016 [15] provides specifications for the use of RTP header fields and fragmentation rules for MPEG-4 audio and video. MIME (Multipurpose Internet Mail Extension) type registration and SDP (Session Description Protocol) [21] are used for the MPEG-4 session establishment. This is useful to configure the decoding parameters associated to a particular ES through the RTP session initialization. In this approach, MPEG-4 audio and video streams are not managed by MPEG-4 systems Object Descriptor but by H.245 session signaling protocol or other out of band means. The BIFS language is also not used to manage the scene that consequently reduces video content manipulation and interaction.

The fragmentation rules described in this document are flexible. It defines the minimum rules for preventing meaningless fragmentation. LATM (Low-overhead MPEG-4 Audio Transport Multiplex) manages the sequences of audio data with relatively small overhead. In audio-only applications, it is desirable for LATM-based MPEG-4 Audio bit streams to be directly mapped onto the RTP packets without using MPEG-4 Systems. This help reducing the overhead caused by the Sync layer.

3.2. RTP payload format with reduced SL header

This approach is represented by the MPEG-4 Generic payload format described in [16] and its companion simplified draft [18]. Both of them are at status of Internet draft and seem to be an alternative for [15]. The approach presented in [18] implements a subset of the [16] features and addresses the AAC (Adaptive Audio Coding) and CELP (Coding Excited Linear Predictive audio) audio streams encapsulation over RTP.

The approach recommends the use of few significant fields of the SL header with an eventual mapping to RTP header and RTP payload header. The RTP payload header (reduced SL Header) is fully configurable and can fit to different ESs types (video, audio, BIFS and ODs). In addition, this payload format can be configured to be compatible with [15].

With this payload format, only a single MPEG-4 elementary stream can be transported over a single RTP session. Information on the type of MPEG-4 stream carried in RTP payload is conveyed by MIME format parameters through SDP message or others means. It is possible to carry multiple Access Units originating from the same ES in one RTP packet.

3.3. RTP Payload Format for MPEG-4 FlexMultiplexed Streams

The approach [17] is a former Internet draft. It recommends encapsulating an integer number of FlexMux packets in the RTP payload. Inversely to other payload formats approaches, this one provides an elementary streams multiplexing over an RTP session and use the complete MPEG-4 terminal specification (i.e. the Sync Layer and FlexMux syntax).

3.4. RTP Payload Format with a Single AU Encapsulation

This approach is presented in [19]. It provides an elementary streams multiplexing scheme over a single RTP session. The multiplexing scheme is achieved through the ES_ID mapping in the RTP header. Each RTP packet encapsulates a single AU originating from one ES. To distinguish among several MPEG-4 streams, the ES_ID of each AU is mapped to SSRC (Synchronization Source) RTP header filed. The MPEG-4 system streams (i.e. BIFS and OD streams) are conveyed through a reliable mean.

3.5. Feature Comparison and Analysis

[15] proposes fragmentation rules for the MPEG-4 video streams that takes care of the MPEG-4 video stream hierarchy such as VS (Visual Sequence), VOL (Visual Object Layer), VO (Video Object), and VOP (Video Object Plane). In [16] and [18] approaches the Sync Layer's configurable syntax allows an alignment with [15] and takes a benefit from the use of certain Sync Layer fields, such as CTS (Composition Time Stamp) and DTS (Decoding Time Stamp) for an efficient intra-media synchronization. Finally, this payload format allows AU (Access Unit) concatenation to improve the bandwidth usage by reducing the overhead. Nonetheless, these approaches transport each ES as an individual RTP session which may be unpractical, unscalable and bandwidth inefficient in wireless multicast communications. Since MPEG-4 scene may contain many ESs, allocating and controlling hundreds of destination address for each MPEG-4 session, it is a complex task for both source and destination terminals. This will definitely limit MPEG-4 service scalability. In addition, for optimizing the good throughput, AU buffering and concatenation is usually done at the source terminal. Consequently, the packetization delays became intolerable for interactive communications. Hence, there is a tradeoff between the wireless bandwidth optimization and the end-to-end communications delay minimization.

On the other hand, the multiplexing scheme provided by [17] allows a reduction of the total RTP sessions. Nevertheless, there are an excess of overhead and redundancy that cannot be removed due to the intercalation of the FlexMux header between the Sync Layer header and the RTP header. Such superfluous overhead is obviously too costly in terms of bit rate, which is intolerable in the wireless network low bandwidth.

The [19] approach provides an efficient multiplexing combined with a low overhead encapsulation (i.e. without the use of the Sync layer). However, the encapsulation of a single AU in the RTP payload may be inefficient and causes an unacceptable overhead when transporting low bit rate MPEG-4 media streams (e.g. CELP voice communications). Typically the low bit rate MPEG-4 streams generates small AU size.

Figure 2 illustrates approaches of encapsulating MPEG-4 stream over IP compared to the ISO MPEG-4 standard approach.

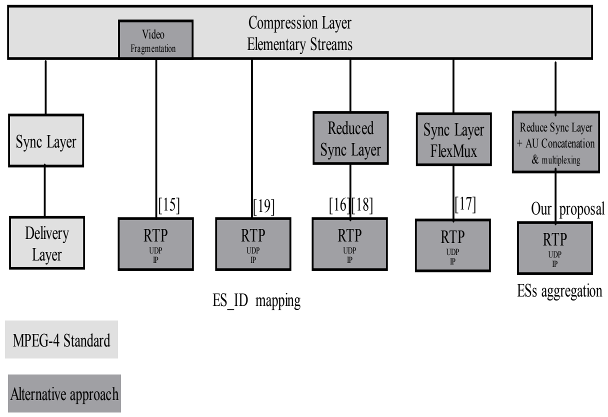


Figure 2: Concurrent approaches for encapsulating MPEG-4 streams over IP.

4. RTP4MUX: A NEW RTP PAYLOAD FORMAT FOR MPEG-4 MULTICAST WIRELESS IP COMMUNICATIONS

Wireless IP networks are characterized by low and fluctuating bandwidth links. In addition, audiovisual applications required stringent QoS guarantees and are hard to control over such networks. Hence, an appropriate data packetization, multiplexing and transport at the source are essential. The packetization process must address the overhead reduction and provides a generic RTP payload format. It has to also offer some Sync Layer features. Furthermore, an efficient ESs multiplexing mechanism is much suitable for transporting multiples MPEG-4 ESs in a single RTP session. This will facilitate the ESs management, optimize the RTP payload use, and favor the MPEG-4 terminal scalability.

An efficient data partitioning and encapsulation scheme must minimize the overhead while transporting small AU size streams. Table 1 illustrates the overhead for common low bit rate MPEG-4 voice and audio formats transported over RTP. It is clear that the AUs concatenation reduces the total packetization overhead.

Audio Object	Audio Payload Length	Overhead (RTP/UDP/IP)	Overhead Rate
AAC (64 kbit/s, 24 kHz)	342 bytes (average)	12+8+20 bytes	10,9%
CELP (6 kbit/s, 8 kHz)	15 bytes (fixed)	12+8+20 bytes	72,7%

Table 1: AAC and CELP audio streams transport overhead.

When AAC (Advanced Audio Coding) is used for encoding of stereo audio signal at 64 kbit/s, AAC frames contain an average of approximately 342 bytes (with 1024 sample/frame). On a Wireless LAN with 1500 bytes MTU (Maximum Transfer Unit), a four AAC frames can be carried in one IP packet. This allows optimizing network resources and bandwidth usage. However, with lower bit rates MPEG-4 streams (e.g. AAC, CELP, Facial Animation, etc.), the interval between a successive AUs generation may be important. Otherwise, the concatenation of multiple AUs from a single elementary stream will introduce a supplementary delay, which is intolerable in a real-time and interactive multimedia communications. For example, the concatenation of 30 CELP AUs will induce 600 ms packetization delay. In addition, since the low bit rate MPEG-4 streams is expected to be largely deployed on the wireless IP networks, optimizing the bandwidth usage trough a good RTP payload exploitation is indispensable.

Within RTP4mux, we propose to aggregate several MPEG-4 elementary streams on a single RTP session. We achieve a concatenation of several AUs, from several ESs, in the same RTP payload (i.e. It is possible to convey several ESs in the same RTP packet). The motivation for packetizing multiple AUs in the same RTP packet is the overhead reduction. While the motivation for concatenating AUs from different ESs is minimizing the end-to-end ESs transmission delays. Furthermore, this aggregation scheme minimizes the dependency between adjacent RTP packets, which mitigate the dependency of MPEG-4 decoder on any lost packet. In our case, the regrouping of particular streams in the same RTP payload is made as follows:

- Based on taking together elementary streams that are tightly synchronized, such as facial animation stream and its associated CELP speech stream, etc.
- Or, based on the Quality of Service considerations (i.e. AUs originating from ESs with the same QoS requirements are grouped together in one RTP packet). This facilitates the QoS mapping over the network.

When transporting each ES in a single RTP session, like used in [15], [16], and [18], the multiplexing of the ESs is achieved through signaling mechanism. ES_ID of each MPEG-4 stream is sent at the RTP session initialization by using the MIME format parameters and SDP messages. Thus, due to the dynamic nature of MPEG-4 scenes with intervention of a new Media Object (ES) during the MPEG-4 sequence. This signaling mechanism involves additional delays not suitable for real-time communications.

Before giving implementation details of the RTP4mux protocol, we describe briefly the principal RTP4mux features in what follows:

Elementary streams fragmentation: RTP4mux takes into consideration the nature of transported streams. In addition, it behaves better against the wireless network losses through an appropriate ESs multiplexing and encapsulation schemes. The fragmentation provides ESs fragmentation into independently decodable entities

Elementary streams multiplexing: It is a two levels ESs multiplexing scheme, which is based on (1) multiplexing several ESs in a single RTP session and (2) encapsulating different AUs (originating from different ESs) in the same RTP payload. This multiplexing scheme provides better WLAN bandwidth exploitation and reduces the en-to-end delays.

Elementary streams encapsulation: It provides a transport mechanism over the RTP/UDP/IP stack. The main purpose of the encapsulation scheme is to allow ESs interleaving which reduces loss impact on the received stream.

Elementary streams synchronization: It is a synchronization based on the MPEG-4 system (i.e. it provides inter and intra ESs synchronization). RTP4mux uses the transport-level synchronization (provided in the RTP header) and the Sync Layer synchronization (provided in the reduced SL header).

QoS provisioning in the WLAN: Based on our previous works ([19],[22] and [23]), we're working on providing dynamic QoS provisioning over a Differentiated Services WLAN through deploying a PHB (Per Hop Behavior) for each group of aggregated streams (i.e. one PHB for each RTP4mux stream).

4.1. Elementary Streams Fragmentation and Packetization

We propose to carry in the RTP payload several reduced SL packets, which are originated from different ESs (see Figure 3). Our packetization mechanism reduces RTP packet loss's consequences at the receiver side. It offers losses tolerance without deploying an AU's interleaving mechanism like used in [18]. This is due to the packetization mechanism which carries several AUs originating from different ESs in the same RTP packet. The packetization process must take care of the path MTU constraints. In our case, the loss of a single RTP packet doesn't involve successive losses of AUs from the same elementary stream. Loss of consecutive AUs from the same ES can cause a poor quality at the player due to the temporal coding dependencies.

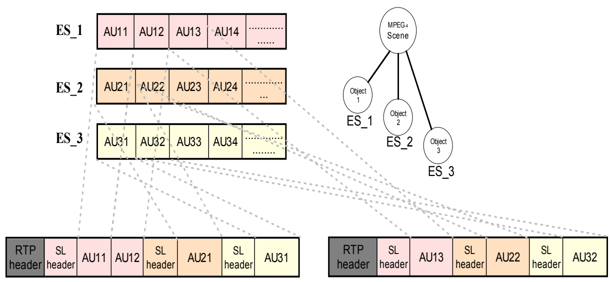


Figure 3: Elementary Streams encapsulation process.

It is clear that our packetization mechanism optimize the bandwidth usage through the overhead minimization by (1) sharing of a single RTP packet header by a multiple reduced SL packets and (2) sharing of single reduced SL packet header by several AUs.

4.2. Elementary Streams Multiplexing

When deploying the object-based MPEG-4 video (i.e. with several Audiovisual ES) over a Wireless IP network, multiplexing is unavoidable due to the large number of ESs that may be used in each MPEG-4 session.

Our proposed encapsulation process involves an ESs multiplexing design, which is performed at 2 levels. First, we provide an MPEG-4 elementary streams multiplexing over a single RTP session. Second, we transport different elementary streams in the same RTP payload. However, this multiplexing scheme involves the identifying of the different AUs that are carried in the RTP payload through their ES_IDs signaling.

In the previous works, [15], [16], and [18], the ES_IDs signaling is achieved at the MPEG-4 session initialization through SDP message. This is possible due to transporting single ES within a single RTP session. In this case, the receiver knows a priori the configuration and the organization of the RTP payload (i.e. presence and size of each RTP payload field). To permit a de-packetization without ambiguity, we have identified two distinct ES_ID signaling approaches.

4.2.1. ES_ID Signaling through the RTP Header

In a similar manner with the FlexMux tools [14] (codeMux mode), we transmit in the RTP packet header a code value that indicates the RTP payload organization. In such a way to make a correspondence between each ES_ID and its associated reduced SL packet, which is carried in the RTP payload. This code field may be mapped into the SSRC RTP header field (see Figure 4). Nevertheless, this approach induces additional out of band signaling of the correspondence tables between the codeMux field and the associated ES_IDs.

In addition, the dynamic behavior of MPEG-4 scene (e.g. apparition of a new ESs during the MPEG-4 session) induces a continuous signaling of the correspondence tables, which are exposed to loss. This will result in a multiplexing blocking, then a decoding blocking.

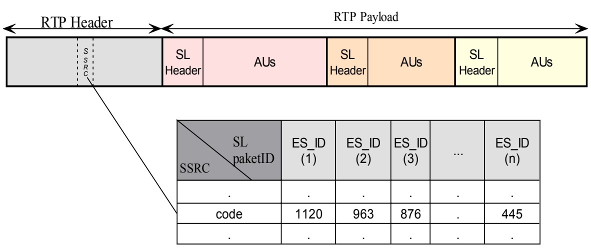


Figure 4: ES_IDs signaling though the RTP header.

4.2.2. ES_ID Signaling through the RTP Payload

This signaling approach is based on in-band ES_ID transmission. In fact, this signaling approach offers a de-packetization without ambiguity through the transmission of the ES_ID in each reduced SL packet (see Figure 5). Otherwise, this mechanism doesn't needs any other out of band signaling stream, which will improve the wireless communication robustness against channel errors and burst packet loss.

This approach will be adopted for the remaining of our RTP payload format definition.

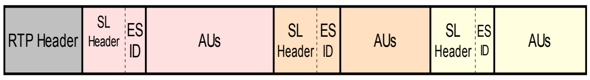


Figure 5: ES_ID signaling through the RTP payload.

4.3. MPEG-4 Elementary Streams Synchronization

Inter and intra-ES synchronization is crucial for deploying audiovisual services over wireless IP networks. Figure 6 shows our applications scenarios including corporate communications such as videoconferencing and video surveillance. The audiovisual equipments, such as camera and microphone associated to a talking person, are connected directly to the RTP4mux Gateway. The RTP4mux gateway provides interleaving, multiplexing, and synchronization of the different ESs into one MPEG-4 stream.

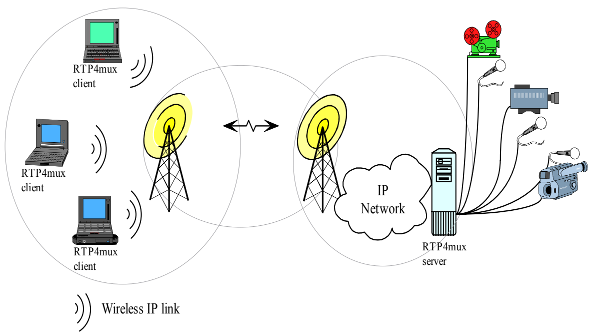
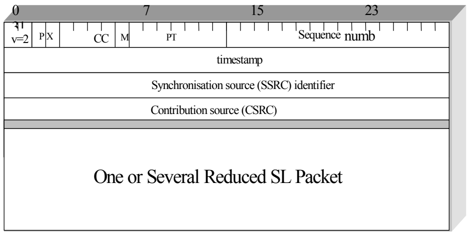


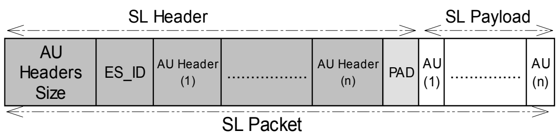
Figure 6: Videoconferencing over wireless IP network using RTP4mux Gateway.

The synchronization used within RTP4mux is derived from the MPEG-4 system (Synch Layer) and implemented in coordination with existing RTP synchronization framework.

RTP Packet format is described in Figure 7.



(a) Header information



(b) Payload format

Figure 7: RTP packet format.

Marker (M) bit: The M bit is set to 1 to indicate that the RTP packet payload includes the end of each Access Unit carried in the RTP packet. As the payload either carries one or more complete Access Units or a single fragment of an Access Unit, the M is always set to 1, except when the packet carries one or multiple single fragment of an Access Unit that is not the last one.
Timestamp: Indicates the sampling instance of the first AU contained in the RTP payload. This sampling instance is equivalent to the CTS (Composition Time Stamp) in the MPEG-4 time domain.
SSRC, CC and ***CSRC*** fields are used as described in the generic RTP header RFC 1889 [7].
We define in the reduced SL packet the following fields (see Figure 7 (b)).
AU Headers Length: is a two bytes field that specifies the length in bits of the concatenated AU-headers. If the concatenated AU-headers consume a non-integer number of bytes, up to 7 zero-padding bits must be inserted at the end (PAD field) in order to achieve byte-alignment of the AU Header Section.
ES_ID: is a two bytes field that specifies the ES_ID associated to the AUs carried in the reduced SL packet. This field is common to all the AUs encapsulated into the SL packet. This minimizes the overhead. The ES_ID field is the only one that must be present in the SL packet header.
For each Access Unit in the SL packet, there is exactly one AU-header. Hence, the nth AU-header refers to the nth AU.

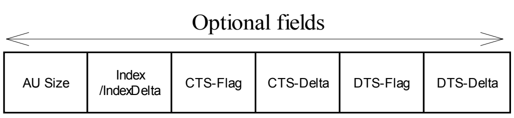


Figure 8: AU header's fields.

We define in the AU Header the following fields (see Figure 8):
AU Size: indicates the size in bytes of the associated Access Unit in the reduced SL payload.
Index / IndexDelta: indicates the serial number of the associated Access Unit (fragment). For each (in time) consecutive AU or AU fragment, the serial number is incremented with 1. The AU-Index-delta field is an unsigned integer that specifies the serial number of the associated AU as the difference with respect to the serial number of the previous Access Unit. The Index field appears only on the first AU Header of the reduced SL packet.
CTS-Flag: Indicates whether the CTS-delta field is present. A value of 1 indicates that the field is present, a value of 0 that it is not present.
CTS-Delta: Encodes the CTS by specifying the value of CTS as a 2's complement offset (delta) from the timestamp in the RTP header of this RTP packet. The CTS must use the same clock rate as the time stamp in the RTP header.
DTS-Flag: Indicates whether the DTS-delta field is present. A value of 1 indicates that the field is present, a value of 0 that it is not present.
DTS-Delta: specifies the value of the DTS as a 2's complement offset (delta) from the CTS timestamp. The DTS must use the same clock rate as the time stamp in the RTP header.

We propose to signal each RTP payload configuration through SDP messages at the RTP session initialization. The appropriate MIME parameters are described in [24].

5. PERFORMANCE EVALUATION

RTP4mux is evaluated with the Network Simulator v2 [25] using various network configurations. We emphasize the robustness and efficiency of our proposal with the classical MPEG-4 streams encapsulations recommended by the IETF proposal [18].

Both Interactive VideoConferencing and One-way VideoSurveillance applications are considered for the simulation as described in Figure 6. In this case and within our proposal, the MPEG-4 server generates two RTP sessions. The first RTP session aggregates the video streams while the second one aggregates the audio streams. The efficiency and robustness of our proposal is validated for the audio streams transmission only. Video stream performances are similar and can be found in [24].

In our simulation, we use a MPEG-4 CELP audio stream (see Table 2) which is dedicated to the speech coding. Based on a predictive coding, CELP streams are very sensitive to consecutive AUs losses. In addition, Since the CELP is a low bit rate audio stream (4 – 24

Kb/s), the packetization delays may cause an intolerable synchronization problems (e.g. echo phenomena's).

Audio Object	Audio Specific Configuration	AU Size	AUs Generation Interval
CELP (2 minutes long)	mono, 6 Kbit/s (fixed), 8 KHz sampling, 160 sample/AU	15 bytes (fixed)	20 ms

Table 2: CELP audio streams parameters.

We take as simulation scenario a MPEG-4 session, which involves multiple ESs transport. The MPEG-4 session represents an audio conference with several audio objects (i.e. several talking persons). The simulation is performed on infrastructureless mobile network, commonly known as an ad hoc network. The mobility, routing, and bandwidth variation constraints are not taken into consideration.

5.1. Network Model

We use the network simulation models depicted in Figure 9 for evaluating and comparing our proposal with the classical approach. Three CELP voice streams are transported over an ad hoc network using both approaches. Also, the network topologies and parameters are similar for both approaches. The ad hoc network is composed of 6 nodes. The wireless links characteristics (i.e. the channel bit rate and the transfer delay) are illustrated in the Figure 9.

The MPEG-4 source terminal attached to the node "1" sends the CELP voice streams to the MPEG-4 destination terminal attached to the node "5". We include constant-bit-rate (CBR) traffic over UDP to make the link between the nodes "3" and "4" congested A CBR sources allows loading the network differently each time in order to get further information about our packetization scheme.

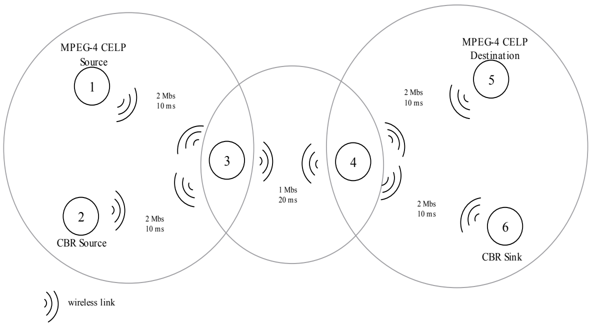


Figure 9: The wireless IP network Model.

In order to optimize the bandwidth usage, we maximize the RTP payload exploitation for both approaches. The RTP packet size is fixed to 210 bytes. This permits encapsulating 12 AUs from a CELP stream into one RTP payload. The encapsulation takes care of the RTP payload header fields (i.e. SL packet header fields) for either [18] or our proposal. In the classical encapsulation algorithm [18], the concatenation of several AUs, from a single CELP stream, into the RTP payload induces the transport of 12 AUs (i.e. 240 ms of voice) per RTP packet. With our encapsulation proposal, we achieve a multiplexing of 3 different CELP streams, which will result with the transport of 4 AUs (i.e. 80 ms of voice) from each multiplexed CELP stream. At the end, we simulate both of the approaches through a network varying conditions. Results are described in Figure 10, Figure 11, and Figure 12.

5.2. Results Analysis

Figure 10 illustrates the end-to-end transmission delay for each AU generated by the CELP coder. This delay was calculated from the AU's coding generation time (CTS) until the AU's decoder reception time. We note that in the Figure 10, our proposal provides a better end-to-end delay transmission (means of 0.075 in our proposal compared to 0.175 of the IETF approach). This is especially due to our RTP packetization process (i.e. multiplexing scheme), which takes only 80 ms while in the IETF approach RTP packetization takes 240 ms.

In the other part, the instantaneous losses measurements (see Figure 11) reveal a better behavior for our proposal. In Figure 12, we experiment the loss of 4 adjacent AUs from a single CELP voice stream per RTP packet lost. While in [18] the RTP packet loss induces the loss of 12 adjacent AUs from the same CELP voice stream. Thus, the loss of 240 ms of compressed stream will lead to devastating consequences at the decoding side.

Also, when interleaving of AUs is deployed with the [18] approach. Both end-to-end delays and losses tolerance remains an immense inconvenience due to the CELP low bit rate, AU small size, and the number of AUs transported into each RTP packet.

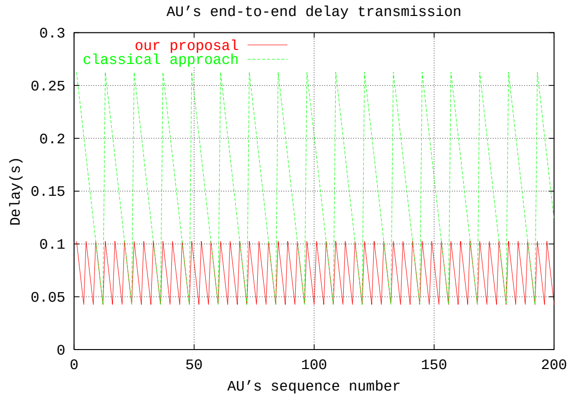


Figure 10: End-to-End AU's transmission delays, calculated between the coding generation time and the decoder reception Time.

Figure 11: The instantaneous AU's loss rate. Calculated, every 1,2 seconds, for all the audio conference duration.

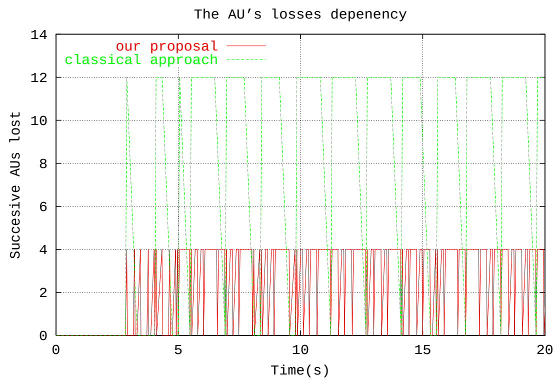


Figure 12: The correlated AU's losses.

6. CONCLUSION

In this paper, we reviewed and compared recent on-going research on the field of transporting MPEG-4 streams over IP networks at the IETF. We then presented and evaluated a novel RTP payload format (RTP4mux) for efficient support of MPEG-4-based multicast communications over low-bandwidth wireless IP networks. RTP4mux provides MPEG-4 data partitioning, multiplexing, and encapsulation over a single RTP session for a significant reduction of control overhead, packetization latency, and better session management scalability. Performance evaluation has shown the robustness of our proposal against channel errors. A significant reduction of the end-to-end transmission delays of MPEG-4 Access Units has been also measured. This is particularly noticeable when low bitrate audio coding such as AAC and CELP are used. RTP4mux achieves both bandwidth efficiency and loss tolerance through a flexible and a configurable ESs interleaving and a two-levels access unit multiplexing mechanism.

7. REFERENCES

[1] B. Crow, I. Widjaja, J.G. Kim and P. Sakai, "Investigation of the IEEE 802.11 Medium Access Control (MAC) Sublayer Functions," in Proc. INFOCOM, 1997, pp. 26-33, vol. 1. 1997.

[2] S. Chen and K. Nahrstedt "Distributed Quality of Service Routing in Ad Hoc Networks" IEEE Journal on Selected Areas in Communications, 17(8), August 1999.

[3] A. Zahedi, and K. Pahlavan, "Capacity of a Wireless LAN with Voice and Data Services," IEEE Trans. Commun., vol. 48, pp. 1160-1170, July 2000.

[4] V. Paxson, "End-to-End Internet packet dynamics", Proc. ACM Sigcomm '97, Cannes, France, Sept. 1997.

[5] M. Podolsky, C. Romer, S. McCanne, "Simulation of FEC-based error control for packet audio on the Internet", Proc. IEEE Infocom'98, San Francisco, CA, April 1998.

[6] J-C. Bolot, "End-to-End packet delay and loss behaviour in the Internet", Prc. ACM Sigcomm '93, San Francisco, CA, pp. 189-199, Aug. 1993.

[7] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson; "A Transport Protocol for Real-Time Applications" RFC 1889 Internet Engineering Task Force, January 1996.

[8] Carsten Herpel. ISO/IEC JTC1/SC29/WG11/N2615, Architectural Considerations for Carriage of MPEG-4 over IP. 1998.

[9] Nicolas G.Feamester; "Adaptive Delivery of Real-Time Streaming Video". Doctorate Thesis; Electrical Engineering and Computer Science, Massachusetts Institute of Technology, May 2001.

[10] ISO/IEC 14772-1 International Standard; "Virtual Reality Modeling Language (VRML)", 1997.

[11] ISO/IEC 14496-1, 01 "Coding of audio-visual objects -- Part 1: Systems -final committee draft"., Version 1: January 1999, Version 2: January 2000; Version 3: January 2001.

[12] ISO/IEC 14496-2, 01 “Coding of audio-visual objects -- Part 2: Video - final committee draft”. Version 1: January 1999, Version 2: January 2000; Version 3: January 2001.

[13] ISO/IEC 14496-3, 01 “Coding of audio-visual objects -- Part 3: Audio - final committee draft”. Version 1: January 1999, Version 2: January 2000; Version 3: January 2001.

[14] ISO/IEC 14496-6, 01 “Coding of audio-visual objects -- Part 6: Delivery Multimedia Integration Framework (DMIF) final committee draft.” Version 1: January 1999, Version 2: January 2000; Version 3: January 2001.

[15] K. Yoshihiro al., " RTP Payload Format for MPEG-4 Audio/Visual Streams", IETF RFC 3016, November 2000

[16] P. Gentric al., "RTP Payload Format for MPEG-4 Streams", IETF Draft, July 20, 2001, expires January 20, 2002.

[17] D.Curet, E.Gouleau, S.Relier, C.Roux, P.Clement, G.Cherry; Internet Draft “RTP Payload Format for MPEG-4 FlexMultiplexed Streams” IETF Draft, November, 8 2001. Expired May 2001.

[18] J. Van Der Meer al., "Transport of MPEG-4 Elementary Streams", IETF Draft, July 2002, expires January 2003

[19] T. Ahmed, G. Buridant and A. Mehaoua, « Encapsulation and Marking of MPEG-4 Video over IP Differentiated Services », IEEE Int. Symposium on Computers and Communications (ISCC’01). Tunisia, July 2001, pp. 346-352.

[20] D. Wu, Y.T. Hou, W. Zhu, Y. Zhang and H.J. Chao “MPEG-4 Compressed Video Over the Internet” Circuits and Systems, 1999. ISCAS '99. Proceedings of the 1999 IEEE International Symposium on , Volume: 4 , 1999

[21] Handley, V. Jacobson, “SDP: Session Description Protocol”, Internet Engineering Task Force, April 1998.

[22] T. Ahmed, A. Mehaoua, G. Buridant, “Implementing MPEG-4 Video on Demand over IP Differentiated Services”, Global Telecommunications Conference, 20001. GLOBECOM’01. IEEE, Volume: pp. 2489-2493 vol. 4. April 2001.

[23] A. Mehaoua, R. Boutaba “A cooperative QoS control framework for streaming video applications”, Global Telecommunications Conference, 20001. GLOBECOM’01. IEEE, Volume: pp. 2470-2474 vol. 4. April 2001.

[24] A. Nafaa, T. Ahmed, A. Mehaoua “Transport Optimization and Flow Control of Real-Time MPEG-4 video Streams Over IP” Technical report PRiSM Lab., University of Versailles, France, Sept. 2002.

[25] Fall, K and K. Varadhan, “Network Simulator note and Documentation”.