Reliable D-Cinema Multicasting over Heterogeneous Networks

Paolo Micanti, Giuseppe Baruffa, and Fabrizio Frescura
Dept. of Electronic and Information Engineering (DIEI), University of Perugia, Italy
Email: {paolo.micanti, giuseppe.baruffa, fabrizio.frescura}@diei.unipg.it

Abstract— **This paper presents a packetization strategy for the reliable electronic distribution of Digital Cinema (D-Cinema, DC) content. In this paper, new experimental results and design improvements are introduced, with respect to those presented in [1]. A reliable multicast protocol was used to send high definition DC video contents to multiple receivers, allowing them to correctly acquire the entire stream. NORM (NACK Oriented Reliable Multicast) was chosen as multicast protocol, because of the guarantees offered on the reliable transmission of data. The target result is to send reliably DC contents, by exploiting a number of multicast-enabled heterogeneous networks, such as fiber, satellite, WiMAX, etc. The results show that a reasonable transfer speed may be achieved even in presence of low transmission bandwidth and moderate packet loss rate.**

Index Terms— **Digital Cinema, WiMAX, wireless networks, multicast, JPEG 2000**

I. Introduction

The technological developments in the fields of High Definition (HD) video capture and projection devices, high-speed data network and storage hardware and advanced digital image compression algorithms, made "Digital Cinema" (D-Cinema, DC) [2] feasible. Large cinema chains worldwide have already opened theatres with digital projection systems, and a big growth in the number of digitally enabled screens is expected in the next few years. Much of the effort for deploying this new technology is due to Digital Cinema Initiatives (DCI) [3], a joint venture of several studios, aiming to establish uniform specifications for digital cinema. This framework does not specify the distribution strategies to be used for content delivery from production sites to theatres, which is left to technological development and to the market: the adopted transmission media could be high-speed wireless transmission, satellite, or magnetic storage.

In Europe, some research projects [4], [5] are addressing the DC delivery chain and they are evaluating both satellite and WiMAX [6] as possible distribution technologies.

A possible operating scenario for D-Cinema distribution is the one depicted in Fig. 1, where pre-recorded HD movies or live events are transmitted from the production/storing center to an head end, from which they are sent, through a network provider, to one or more regional theaters (Multiplex), which in turn could act as proxies for smaller theatres located in sub-urban or rural areas. Even direct user access to video feeds is envisaged, if a network connection with an appropriate bit rate is available, e.g., satellite, WiMAX, or HDSL. Some research projects and trials have showed the feasibility of video broadcasting over high speed wireless connections [7], [8].

The distribution of DC contents to the projection site is one of the many problems that must be addressed in order to spread this new technology. The basic solution of directly distributing the movies with hard discs or high capacity optical media is always viable, but it could be more efficient to adopt a delivery method in which all the contents are electronically transferred from the distribution headquarters to the theaters.

In this case, DC content may be delivered using several communication channels, and many of them could also be wireless, such as, for example, DVB-T [9]

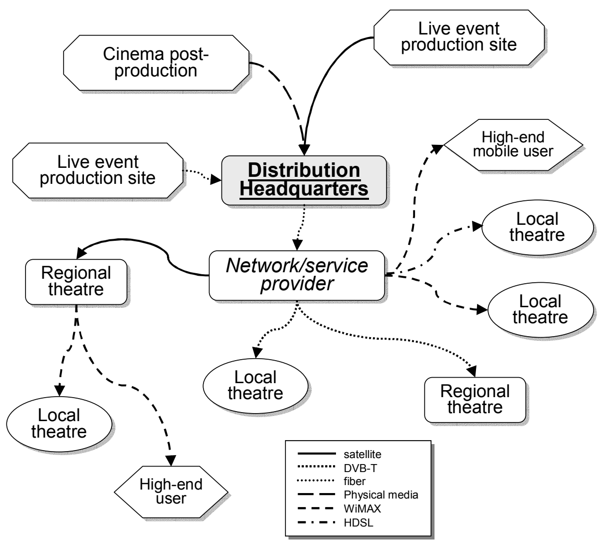

Figure 1. Possible distribution scenario for pre-recorded movies and live events.

Based on "A Packetization Technique for D-Cinema Contents Multicasting over Metropolitan Wireless Networks", by P. Micanti, G. Baruffa, and F. Frescura, which appeared in the Proceedings of the IEEE International Conference on Mobile WiMAX 2007, Orlando, FL (USA), March 2007. © 2007 IEEE.

This work was supported by the European Union under the project "EDCINE", IST-038454, http://www.edcine.org.

and WiMAX. With wireless technology, the destination of the contents can also be a mobile user: for instance, an audience located on a train or on a bus. Bandwidth requirements represent the fundamental constraint for the transmission of digitized video content: due to the very high bit rate required (for HD raw video, it can be in excess of 5-6 Gbit/s), lossy compression of the video stream is required, and the emerging JPEG 2000 standard [10] has been chosen as preferred source coding method, due to its higher image quality and scalability properties, when compared to other similar existing systems. As a result of the compression, the final bit rate drops down to 250-500 Mbit/s.

WiMAX may become one of the preferred wireless distribution systems. The transmission to theaters cannot yet happen in real-time: current technological limitations and the relative novelty of the system impose a downlink speed limit in the order of 10 Mbit/s, at least for the next few years. In addition to transport and content delivery to regional and local theaters, the same WiMAX network, with meshing applied [11], can also be used to deliver content to high-end customers, directly feeding personal home theaters.

Nonetheless, a reduction in the audio and video quality requirements (which, in turn, means a lower bandwidth) enables the use of real-time streaming techniques for applications such as live events transmission and (Near) Movie-on-Demand; however, additional methods for the decrease of bit rate should be adopted, such as interframe coding (i.e., exploitation of the temporal redundancy between adjacent video frames) or higher compression factors. When JPEG 2000 video has to be delivered over wireless IP networks, there are some problems that must be challenged [12]. In the case of TCP, for example, a nonconstant transmission delay is deleterious for many real-time applications.

In short, when dealing with the transfer of DC over a heterogeneous environment with different types of networks, the requirements can be specified in the following terms:

- limited bandwidth, not only for the compressed content itself, but also considering some signaling overhead due to the adopted transfer or streaming protocols;

- reliable transmission of contents, by use of Forward Error Correction (FEC) techniques and/or Selective retransmission of lost data (Selective ARQ).

One of the proposed solutions is that of adopting a reliable multicast protocol [13], able to cope with the high bit rate requirements and low packet loss rate needed for the distribution of DC content. In case of non real-time transfer, such reliability should guarantee the sender that every receiver correctly obtains an exact copy of the original data. The NORM (NACK Oriented Reliable Multicast) protocol [14] may represent a solution to this problem: by means of negative acknowledgments sent from receivers to the sender, this protocol achieves reliability. It is conceived on top of the IP multicast layer, thus allowing the use of every multicast-enabled network. In addition to the reliability, offered by the feedbacks, it also improves the transmission by using FEC packets sent along with data, in order to decrease feedback and re-transmission needs, at the expense of additional overhead for parity data.

In this paper, we present a technique for encapsulating D-Cinema compressed video content into the existing NORM protocol packets, thus minimizing the probability of re-transmission by a judicious way to split, send, and re-compose JPEG 2000 data packets. In section II, the technical requirements for DC target quality will be discussed; in sections III and IV, an overview of the adopted multicast protocol and the description of the encapsulation technique will be presented, whereas in section V the global system architecture is described and, eventually, in section VI, the obtained results are presented.

II. Technical Specifications For D-Cinema

A. An overview of JPEG 2000

JPEG 2000 is a source coding algorithm used for image (and video) compression [15], adopted as an international standard by ISO/IEC. This standard offers many advantages, compared to JPEG and other similar compressed formats. Some of its features are:

- at the same comparable image quality, an higher compression ratio can be achieved, thus lowering the bit rate requirements;

- capability to embed, in the same format, a lossy and lossless compressed version of the original image;

- different spatial resolutions, components, and image qualities (*layers*) can be embedded and progressively transmitted/decoded;

- different portions of the codestream can be quickly accessed, also obtained by subdividing the image into smaller, independently encoded sub-images (*tiles*); the basic elements of a codestream are the JPEG 2000 data *packets* (not to be confused with network packets), which represent a set of compressed data sharing some spatial, component, or quality characteristic;

- one or more *regions of interest* (i.e., foregrounds) can be enhanced over the general image background; they represent parts of the image which should be carefully encoded, with more detail than the rest;

- possibility to integrate error resilience and protection for the transfer over noisy environments (such as the case of wireless transmission), either by means of embedded error resilient source coding or added FEC redundancy (JPWL extension, [16]);

- possibility to introduce different methods for integrated security and encryption (JPSEC extension [17]).

The JPEG 2000 standard is different from JPEG and other image compression formats, since it applies the transform coding step (DWT, Discrete Wavelet Transform) over the entire image, rather than on pixel blocks: this represents a main difference with the DCT (Discrete Cosine Transform) used in the JPEG standard (operating on 8x8 pixel blocks). This difference can be observed by comparing two images compressed with JPEG and JPEG 2000: at high compression rates, the first one will show many artifacts (i.e., blocking artifacts), due to the high compression ratio in the 8x8 pixel blocks. In the JPEG 2000 case, instead, the image will show a uniform contour fuzziness, with less visual disturbance. The DWT consists of a digital spatial filtering, both along rows and columns, which concentrates the image energy in the lower frequencies; subsequent applications of the filtering, called *decompositions*, will produce data in the low and high spatial frequencies.

A typical compression process consists in the optional splitting of the original, multi-component image into tiles, which are rectangular regions that are independently encoded (Fig. 2-a). The component data are then filtered according to the bi-dimensional DWT, in order to obtain different resolutions embedded into a single image. The resulting *subbands* (Fig. 2-b) are then decomposed into *precincts* and *codeblocks*, which are small regions of transform coefficients. Each codeblock is then scanned in a particular order and its pixel values are quantized.

After the quantization, data are compressed using an arithmetic coding algorithm. First of all, data are ordered by the EBCOT (Embedded Block Coding with Optimal Truncation) algorithm, which then feeds the arithmetic coder. This is a source coding algorithm that uses symbol probabilities to reduce the size of the symbol sequence.

Compressed data will then be used in the rate control process that, depending on the compression parameters (e.g., quality, compression ratio), chooses the optimal amount of data to assemble in the final codestream.

Generally, the data bitstream is represented by a sequence of compressed packets: a JPEG 2000 packet contains all the compressed codeblocks concerning a precinct of a resolution of a component.

Moreover, if a quality layering approach is adopted, each single quality enhancement layer fits into a packet.

The result of the process is a JPEG 2000 codestream, which can be syntactically represented by the structure shown in Fig. 3-a. In the simplest case, a codestream is composed of a main header followed by a sequence of tile parts; the different syntax elements are signaled by a two-byte long *marker*. The codestream always begins with the SOC (start of codestream) marker, followed by the main codestream header (indicated by *MH* in the figure). The main header has some mandatory markers: SIZ, which specifies image information such as image

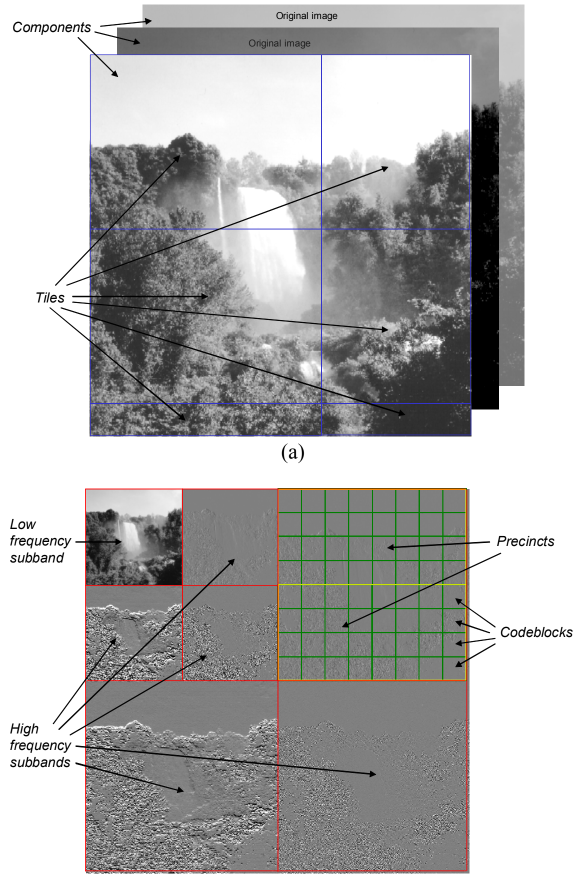

Figure 2. DWT decomposition used in the JPEG 2000 system. The image can be first split in tiles (a) and then decomposed into subbands, precincts, and codeblocks (b).

and tile size, COD (coding style defaults), and QCD (quantization default), which provide default coding and quantization parameters.

The header may also contain some other optional markers, such as for overriding the default parameters on a component basis, or for indicating the regions of interest. A special marker, COM (comment), can be used to add unstructured data to the codestream (comments regarding the image, classification data, author names, etc.). The tile part header consists of two mandatory markers, SOT (start of tile) and SOD (start of data), and some optional markers used to override default coding parameters; the SOD marker is followed by the actual data bitstream. Tile parts may contain either the data

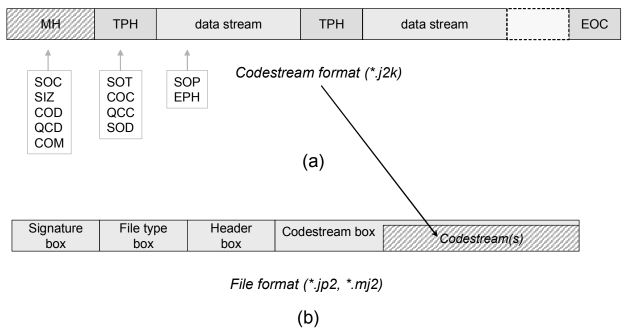

Figure 3. JPEG 2000 codestream marker (a) and file format box (b) structures.

relevant to different tiles of the image, or groups of data relevant to a single tile, which are split according to some pre-defined policy (for example, the tile parts could contain the data of each component, separately). Finally, each JPEG 2000 codestream is closed by the EOC (end of codestream) marker.

JPEG 2000 codestreams can also be wrapped into a special file format, which is useful for adding metadata to the image file itself, such as information required to correctly decode the file, copyright terms, etc. Figure 3-b shows an example of such a wrapped file. A typical, simplified structure uses a sequence of *boxes* that encapsulate all the data. After the signature box, which identifies the file as being part of the JPEG 2000 family, there is the file type box, which specifies the content of the file, followed by the header box, which contains generic information about the file, such as the number of components, color space and resolution, and, finally, by the codestream box, which contains a JPEG 2000 codestream, as defined before. A similar approach is used for motion JPEG 2000 video files, which use the same basic format, but contain more codestreams and additional information boxes.

The knowledge of the structure of a JPEG 2000 codestream and wrapped file can be very helpful when sending it through a network: indeed, it can be easily parsed, to find marker and box boundaries and to classify the bitstream in data packets that can be adjusted to fit the size of the network packets.

JPEG 2000 has been chosen, by the DCI organization, as the compression standard to be used for the lossy coding of DC sequences. Differently from video standards such as H.264 [18] or MPEG-2 [19], the temporal redundancy existing between adjacent frames is not exploited: a DC video file is a collection of ordered codestreams, as well as some additional audio and data tracks, packaged in MXF format [20]. The other possible solution (not envisaged by the DCI specifications) is to use the native Motion JPEG 2000 format [21].

B. Digital Cinema Initiatives system specifications

There are two different profiles specified and covered by DCI recommendations: 2K and 4K. The former profile has an image resolution of 2048x1080 pixels per frame, whereas the latter has 4096x2160 pixels per frame [3]. Both profiles require a precision, for each one of the three color components, of 12 bits, thus totaling 36 bits per pixel. The DCI image structure is required to sustain a frame rate of 24 Frames Per Second (FPS), but it can also support rates of 48 FPS, for the 2K profile only. The color space used for representing the components is the XYZ color space [22], which offers a broader range of color tones over the classic RGB.

As mentioned before, the DC system uses JPEG 2000 coding to achieve a compression level that is visually lossless, in order to limit transmission bandwidth or media storage usage, and it adopts a hierarchical data structure, for a better exploitation of current technological limitations. The JPEG 2000 compression rate is of about 6-7 times: each frame contains exactly one tile, 6 resolutions (5 decomposition levels) for each color component, and a single quality layer; the maximum size of a codestream is of 1,302,083 bytes, for the 24 FPS profiles (this corresponds to a final, encoded video bit rate of 250 Mbit/s). In particular, the data section in the 2K codestream is composed by a sequence of three tile parts, with each part carrying one color component. Since interframe coding is not performed, the video stream is simply a sequence of compressed images. The image compression reduces the required bandwidth from a maximum of 7.5 Gbit/s for a 4K video sequence to 250 Mbit/s.

The management of digital copies is guaranteed by a strong level of encryption, too, in order to protect audiovisual contents from unauthorized duplication and illegal distribution; this is a crucial requirement in wireless networks. In particular, the AES cipher, operating with a 128 bit key, is used to encrypt image and audio data [3].

Fig. 4 shows how DCI movies are prepared and packaged. Edited material represents the Digital Cinema Distribution Master (DCDM): video, audio, and ancillary data. The video content is compressed according to the specifications, and then it is encrypted using the AES algorithm. Audio channels are multiplexed together with the video stream; usually, audio data are not compressed at all, to provide users with an optimal sound experience. During this step, subtitles and captions, in different languages, are added to the content; auxiliary data, which may be used for classification or for projection purposes, can be inserted as well in the final assembly.

This is called the Digital Cinema Package (DCP); it is associated with the keys required for content decryption, which are created by the security manager for every part of the final content: as already mentioned, a DCP is saved in a MXF file.

The DCP is then sent to the projection site: DCI specifications do not define the transmission method, the choice of which is left to marketing or technological considerations. Great attention is paid to security matters: for example, decryption keys are transmitted in a

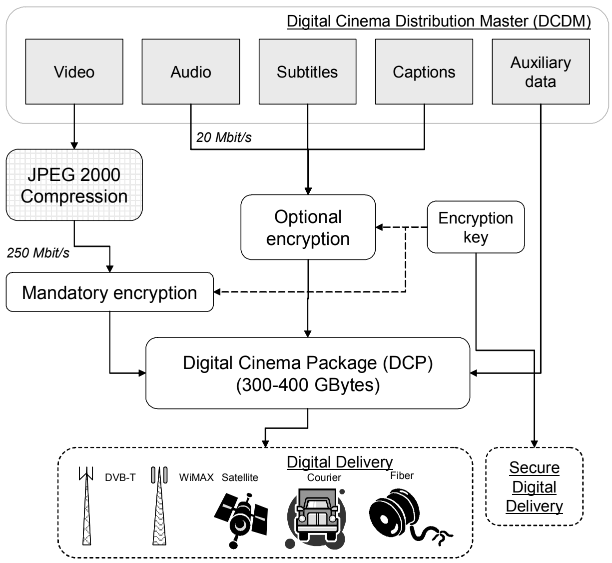

Figure 4. DCI encoding, encryption, and packaging process.

particularly secure way, and the data decoded during projection are on-the-fly decompressed in a secure hardware environment.

III. Overview of the NORM Multicast Protocol

The technical considerations presented in the previous sections, lead to the choice of an efficient method to be used for the transfer of DC content between the distribution headquarters and a number of destinations. Since IP networking is adopted, multicast is a viable solution to the problem.

In this sense, NORM (NACK Oriented Reliable Multicast) [14] is a multicast protocol designed to provide end-to-end reliable transfer.

This protocol uses the generic multicast capabilities of IP, and on top of that it obtains reliable transfer using NACKs (Negative ACKnowledgments). It can work either on reciprocal multicast networks (i.e., wireless or wired LANs) or over unidirectional links (such as unidirectional satellite transmission); in this way, it can satisfy all of the transmission needs for D-Cinema distribution.

The repair capabilities are based on the use of negative acknowledgments, sent from receivers to the sender upon detection of an erroneous or missing packet. The sender transmits packets of data, segmented according to a precise strategy, each one of them being identified by a number.

Whenever a receiver detects a missing packet, it initiates a repair request with a NACK message. Upon reception of NACKs, the sender prepares appropriate repair messages, using FEC blocks. Each receiver can re-initiate a repair procedure if it does not receive repair blocks.

Feedback congestion is a well-known drawback for such techniques: feedback suppression is applied using a random back-off algorithm. This way, each receiver, before sending a NACK for a certain packet, waits for a random time interval, during which it senses the medium and checks if other receivers have issued a repair request for the same packet.

In this case, it discards the NACK; otherwise, it sends the NACK and waits for the repair bits. When the sender receives the negative acknowledgement, it prepares the proper repair bits (or the entire lost packet). Feedback suppression works efficiently in this protocol, and can achieve good results [23].

NACK packets can be sent both in multicast or unicast mode, using the sender address. In the second case, feedback suppression can be achieved using multicast advertising messages, sent from each sender, which let the receivers know which packets have a pending repair request.

A NORM sender can even autonomously add FEC parity bits to each packet, thus enabling the receiver to correct errors and recover from losses, without starting a NACK procedure. FEC parity bits are created using a Reed-Solomon code [24]: the parameters n and k can be chosen in order to accommodate for variations in the channel conditions. The amount of FEC bits to send can also be statically decided in advance.

Fig. 5 shows the typical sequence of operations performed by a NORM sender and receiver during a transmission session.

The sender prepares data packets, segmented according to some parameters that can be changed by the user, to satisfy particular needs: in our case, JPEG 2000 codestreams are encapsulated and an additional header is included at the end of the NORM packet header. It also periodically prepares control messages, such as round trip time collection and rate congestion control feedback.

Each receiver controls if the packet is in order and error-free: in this case, it accepts the packet and forwards it to the destination application. Otherwise, it enters the NACK procedure: this consists in picking a random back-off interval, based on some parameters, such as the largest round trip delay (usually supplied by the sender), and delaying NACK transmission until this interval is elapsed.

In the meanwhile, if it senses a repair request or the repair bits for the same packet, the NACK is dropped; otherwise, it sends the NACK in multicast mode. Sending the NACK in multicast is useful for feedback suppression, as described above.

When the sender receives the NACK, it suspends usual data transmission and immediately sends the repair packets. With these repair bits, receivers are able to recover from transmission errors. If a receiver loses a repair packet, it can resend another NACK for the same packet, after waiting for a new back-off interval.

In particular, senders and receivers can be created, and their behavior can be customized by setting parameters such as transmission speed, TCP port numbers to use, and Reed-Solomon code properties. The embedded congestion control mechanism can be enabled or disabled

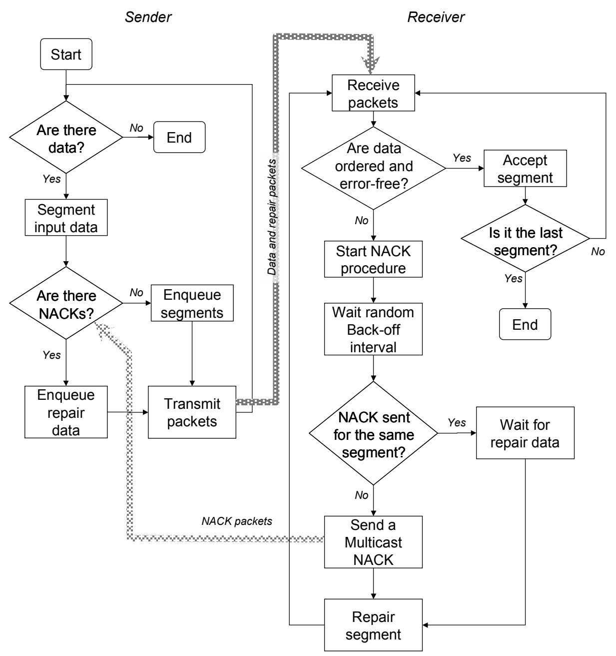

Figure 5. Sequence of operations for the NORM protocol, shown at the sender and receiver sides.

as well.

This protocol can be applied not only to the off-line file transfer between the distribution and destination sites, but it can also represent a feasible method for broadcasting live events to the theaters, in a manner which is similar to that adopted by classic streaming techniques, such as RTSP [25].

In fact, the widespread adoption of WLANs is fostering the diffusion of streaming protocols based on UDP, such as RTP/RTCP, which provide an acceptable performance in case of nonguaranteed packet delivery. Recently, RTP for JPEG 2000 has been introduced and it is in the process of standardization [26].

IV. Packetization and Format for JPEG 2000 Contents

It is important that the sender and the receiver jointly minimize the probability of retransmission; this is achieved by exploiting the underlying codestream-based file structure. A number of fields have been added to the NORM protocol header, based on the fields used by the JPEG 2000 RTP streaming protocol.

A tabular view of the additional header fields is presented in Fig. 6; their meaning and use is described in the following:

- *main_version* (8 bits): it is always set to 0xCB. If the received packet has a different value for this field, it is discarded. This is used to identify all the packets formatted for JPEG 2000 transmission over NORM;

- *version* (8 bits): minor version of the technique, currently 1. Any packet with a different minor version is discarded by the receiver. This guarantees the compatibility between different versions of the packetization method;

- *type* (8 bits): it indicates the kind of transfer, in particular it is related to the format of the codestream-containing file. It can be set to 0xF1 when a set of *.j2c*/*.j2k* files (raw codestreams) is sent, or to 0xF6 if a single *.mj2* video file (wrapped file format) is being transferred. Other types would result in packet discarding;

- *packet_ID* (8 bits): if a single codestream is fragmented at the sender, it indicates the progressive number of the transmission. This is especially useful when transmitting large codestreams, which can be divided into smaller portions at the data packet boundary. This is used for packet re-ordering at the receiver;

- *secondary_ID* (16 bits): it can be used to organize different tracks inside a video file, or to differentiate among video, audio, and synchronization data;

- *image_number* (32 bits): progressive number of an image (i.e., codestream) in the sequence. This is used to reconstruct the video file in order, and to check for the correct reception of packets;

- *offset* (64 bits): it represents the offset of the first data in the current packet, starting from the beginning of the original file; it is expressed in bytes. A field of 64 bits is used to handle files larger than 4 GB (DCPs can be as large as 300 GB). The receiver reads the value of this field for correctly placing the received packet in the destination file;

- *image_length* (32 bits): it is referred to the total length of the codestream containing the image, expressed in bytes. It is used to check if the image has been completely received.

This header is added, for identification, to each sent multicast packet. This packetization strategy can send either separate codestreams or a single video file. In the first case, the content of a directory containing thousands of files (24 or 48 for each second of movie time) is sent, each file having a maximum size of 1.3 MB.

On the other side, we can also send a single video file, which may be larger than 250-300 GB. In this case, we parse the video file and send each codestream in a separate packet.

Fig. 7 shows the sequence of operations performed by the system, before sending the codestream using the NORM protocol.

A Motion JPEG 2000 (MJ2) video file, which contains codestreams and synchronization data in a boxed structure, is parsed to find codestream offsets and lengths. The system allows parsing codestream markers, too: this secondary parsing can be useful if an improved control over FEC data is needed.

V. System Architecture

Fig. 8 shows a typical architecture of a multicast distribution system used for live events. A similar multicast system is also available in case of transmission of a stored video sequence from a production site to theaters and end-users, which is depicted in Fig. 9. Digital cameras capture the event directly to a digital support; such cameras are already available at high resolution (2K and 4K).

The captured video is transmitted to a production site

Bits 7	15	23	31
main_version	*version*	*type*	*reserved*
reserved	*packet_ID*	*secondary_ID*	
image_number			
offset			
offset			
image_length			

Figure 6. JPEG 2000 additional packet header format.

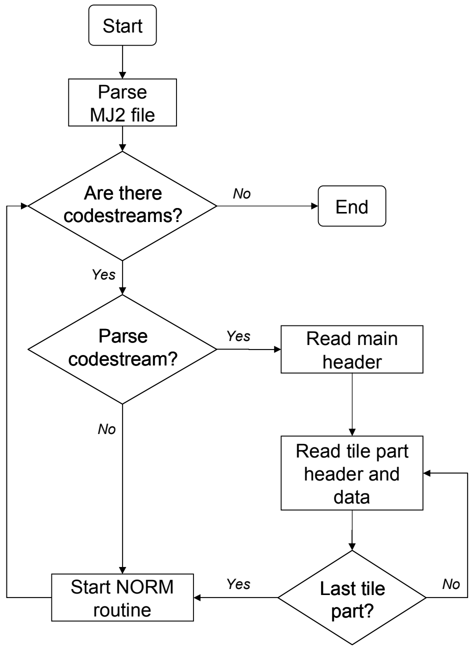

Figure 7. JPEG 2000 codestream parsing operations.

using a high speed network, such as a satellite-based one. In case of a live event, the video content will be immediately sent to the head-end through a high speed network infrastructure and then to the final receivers: they could be large regional theaters, possibly enabled to act also as distribution centers, sending content to local theatres or high-end users. In the last step of the distribution chain, video content can be easily delivered via a wireless channel (WiMAX), also because in rural areas a high-speed wired connection is not always available. If a sufficiently high transmission bandwidth is available, the content could be transferred through the Internet, using a multicast-enabled network or some sort of tunneling system.

VI. TEST APPLICATION AND RESULTS

In our tests, we used a wired architecture, with a server acting as sender and few receivers; they were all connected using a 100 Mbit/s LAN. Fig. 10 shows the architecture of our test system.

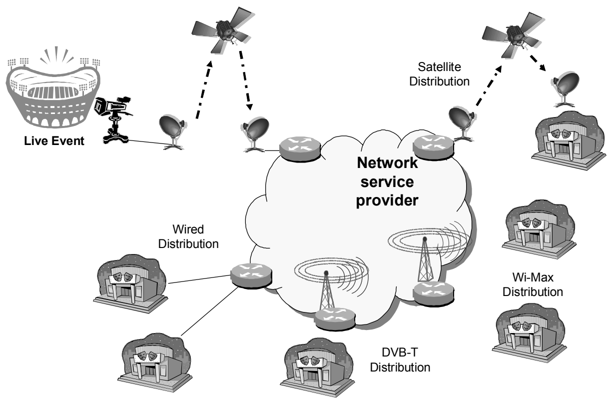

Figure 8. Live video distribution system architecture.

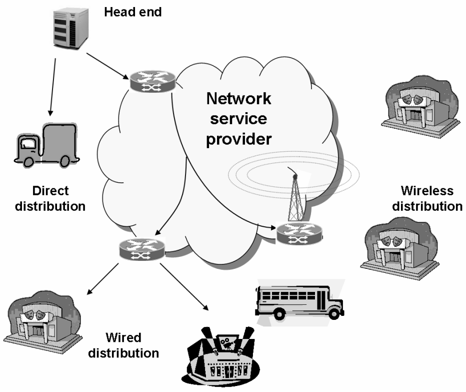

Figure 9. Pre-recorded movie distribution system architecture.

Two separate applications have been prepared for sending and receiving the data.

In order to simulate the presence of transmission errors in the network, we are provided with the capability to generate random errors (at a specified rate) directly either at the sender or the receiver side.

We used workstations with an Intel Core Duo 2 E6600 CPU, 2 GB of RAM, and 300 GB SATA hard disk.

We have performed the tests by sending raw data through the wired LAN, in order to verify the correct operation. The NORM multicast protocol was able to send data, up to a 15% error rate, without too much latency or overhead.

The next step was that to simulate the transmission of a DC sequence; we approximated this by sending DCI-like formatted codestreams, compressed starting from a raw HD sequence (*YCbCr* color space, 4:2:0 chroma subsampling, 2048x1080 pixels, 8 bits per pixel, 25 FPS [27]) adapted to the DCI image quality parameters.

During the tests, transmission errors have been purposely inserted in the sending process: in this case, a small added latency has been observed, and the average speed was reduced due to the NACKing process and packet retransmissions.

Indeed, a very little signaling overhead was introduced by the protocol. In order to compensate for these errors, we took advantage of the FEC capabilities of NORM. Thus, a number of repair bits were introduced in every

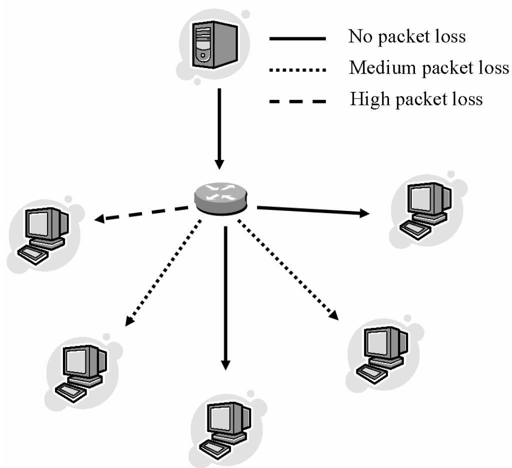

Figure 10. Test system architecture with packet loss indication.

sent packet; in this way, we have managed to reduce the need for NACKs.

Table 1 reports some experimental results found on the 100 Mbit/s wired LAN. The *redundancy bits* column indicates the parameters of the Reed-Solomon code, used by the system. In this experiment, a RS(80,64) code was adopted, where 16 parity bytes are stored in the sender for every 64 data bytes message, and they can be used in case of a NACK; however, the protocol allows sending them *a priori*, thus lowering NACK transmissions and making the data transfer more robust. The *error rate* columns indicate the percentage of errors introduced to simulate lossy data reception.

The results show that the *a priori* transmission of parity bytes is able to cut down the NACKing process with only a small penalty in maximum achieved bandwidth. Fig. 11 shows a graphical comparison of the transmission bit rate, for the two different transmission scenarios. We used the same network architecture with the same packet loss rate, but in the *stored* case the sender calculates, but does not send, FEC bits (they are stored for an eventual future use); in the *sent* case, the application adds a small amount of parity bits to the actual data.

As stated above, sending FEC parity in advance can lower the transmission bit rate, but it will also lower the re-transmission in case of errors. As the figure shows, in case of high packet loss networks (i.e., wireless), the performance achieved by sending parity bits can be higher than that obtained in the normal case.

We are also able to simulate errors at the receiver side, to test the case when, for example, receivers are not close among them, and errors can be assumed to be

TABLE I.
MEASURED TRANSFER SPEED (IN MBIT/S) VERSUS ERROR RATE AND FEC TRANSMISSION METHOD.

Redundancy bits	**Error rate**			
	0 %	***5 %***	***10 %***	***15 %***
stored	81.9	67.1	62.9	52.0
sent	74.1	64.3	63.0	58.9

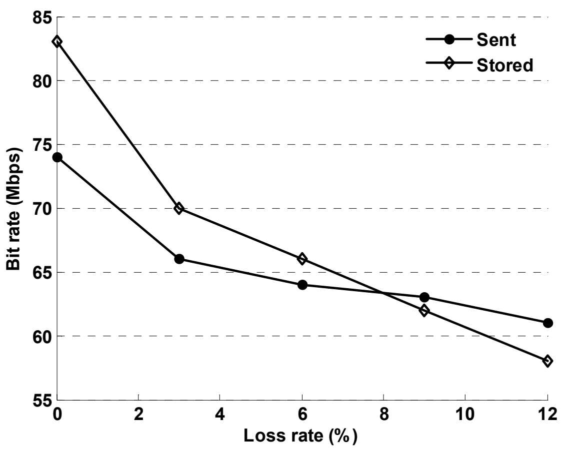

Figure 11. Data rate for *stored* and *sent* transmissions at different error rates.

independent.

We can assume that programmatically introduced errors are independent; in such case, if $p_{e,n}$ is the probability of the E_n error event for the n-th receiver, the probability that the same packet is lost in every receiver is given by the joint probability of the event, that is

$$E_{J(1,\ldots,n)} = E_{e,1} \cap E_{e,2} \cap \ldots \cap E_{e,n} \quad . \tag{1}$$

The errors are simulated by using a uniform random distribution. From (1), we derive the approximated value for the joint probability of packet loss as

$$p_{J(1,\ldots,n)} \cong p_{e,1} \cdot p_{e,2} \cdot \ldots \cdot p_{e,n} \quad . \tag{2}$$

The total error probability, p_{tot}, is computed by assuming that every NACK is received by the sender, that the feedback suppression algorithm manages repeated NACKs, and that there are no errors in receiving the FEC packets. In this case, we can add the single receiver error probabilities, and subtract all the joint probabilities indicating a common packet loss between two receivers. We assume that common losses between more than two receivers are unlikely, due to the generally low adopted loss probabilities. Thus, we obtain

$$\begin{aligned} p_{tot} &\cong p_{e,1} + p_{e,2} + \ldots + p_{e,n} \\ &\quad - p_{J(1,2)} - p_{J(1,3)} - \ldots \\ &= \sum_{r=1}^{n} p_{e,r} - \sum_{r=1}^{n} \sum_{t=r+1}^{n} p_{J(r,t)} \end{aligned} \tag{3}$$

From above, the joint probabilities tend to zero, especially for low error rates. Thus, the total error probability becomes

$$p_{tot} \cong p_{e,1} + p_{e,2} + \ldots + p_{e,n} \quad . \tag{4}$$

Results in Table 2 show the good performance of the protocol in case of high packet loss rate with different groups of receivers. As stated above, for a 9% error rate in two different groups, the total loss rate is about 18%.

VII. CONCLUSION

In this paper, a technique for encapsulating Digital Cinema compressed sequences into a reliable multicasting protocol was presented, for the purpose of distribution among a main production site and the projection theaters by means of heterogeneous transmission networks, including fiber, WiMAX, HDSL, etc. The selected compression standard, JPEG 2000, matched to the HD quality of video sequences, deserves particular care when delivery must also minimize the probability of errors and, consequently, of retransmissions.

Thus, a packetization strategy has been borrowed from the existing RTP strategy for JPEG 2000, and adapted to this particular case. The results, coming from our simplified testbed, have shown the good efficiency of the protocol, even in presence of a moderate/medium rate of transmission errors.

A pre-recorded movie will always be transmitted off-line; thus, even though the channel is error-prone, the reliable multicast protocol (using a high FEC rate) will enable the delivery of the video content to the receivers.

As a future work, more sophisticated techniques for managing channel errors, relying on the tools offered by the JPWL standard, will be introduced and tested. Another possible development of the system is to adaptively modify the FEC rate while running; in this way, temporal variations of the channel loss rate can be properly taken into account.

ACKNOWLEDGEMENT

The authors wish to thank Brian Adamson for his support in the development of the multicast application.

TABLE II.
MEASURED TRANSFER SPEED (IN MBIT/S) VERSUS ERROR RATE AND FEC PERCENTAGE, FOR ONE SENDER AND TWO GROUPS OF RECEIVERS

Error rate for type 1 recs. (%)	Error rate for type 2 recs. (%)	FEC percentage (%)	Average bitrate (Mbit/s)
0	0	0	64.7
3	3	0	47.3
6	6	0	40.8
7	8	0	39.6
8	2	0	43.3
8	8	0	38.1
9	9	0	36.7
10	10	12.5	43.5
15	15	12.5	35.5
20	20	12.5	27.6

REFERENCES

[1] P. Micanti, G. Baruffa, and F. Frescura, “A Packetization Technique for D-Cinema Contents Multicasting over Metropolitan Wireless Network,” in *Proc. of IEEE 2007 Mobile WiMAX Symposium*, pp. 120-124, Orlando, FL, March 2007.
[2] T. Yamaguchi, M. Nomura, K. Shirakawa, and T. Fujii, “SHD Movie Distribution System Using Image Container with 4096x2160 Pixel Resolution and 36 Bit Color,” in *Proc. of ISCAS 2005*, pp. 5918-5921, Kobe, Japan, May 2005.
[3] Digital Cinema Initiatives, LLC, “Digital Cinema System Specification V1.1,” May 2007. Available: http://www.dcimovies.com.
[4] Specific Targeted Research Project, IST call 2, “Layered Compression Technologies for Digital Cinematography and Cross Media Conversion,” *WorldScreen*, 2004. Available: http://www.worldscreen.org.
[5] EDCine Project, IST 6th framework program of the European Commission. Available: http://www.edcine.org.
[6] WiMAX Forum, “Mobile WiMAX – Part I: A Technical Overview and Performance Evaluation,” 2006. Available: http://www.wimaxforum.org.
[7] O. I. Hillestad, A. Perkis V. Genc, S. Murphy, and J. Murphy, “Delivery of On-Demand Video Services in Rural Areas via IEEE 802.16 Broadband Wireless Access Networks,” in *Proc. of 2nd ACM Workshop on Wireless Multimedia Networking and Performance Modeling, WMuNeP*, (jointly with *MSWiM*), October 2006.
[8] Intel Corp., “WiMAX Potential Premieres at Sundance Film Festival,” 2006. Available: http://www.intel.com.
[9] ETSI, “Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for digital Terrestrial television (DVB-T) - EN 300 744 V1.5.1,” June 2004. Available: http://www.etsi.org.
[10] D. T. Lee, “JPEG 2000: retrospective and new developments,” *Proc. of the IEEE*, vol. 93, no. 1, pp. 32-41, January 2005.
[11] M. J. Lee, J. Zheng, Y.B. Ko, and D.M. Shrestha, “Emerging standards for wireless mesh technology,” *IEEE Wireless Communications*, vol. 13, no. 2, pp. 56-63, April 2006.
[12] F. Frescura, C. Feci, M. Giorni, and S. Cacopardi, “JPEG2000 and MJPEG2000 Transmission in 802.11 Wireless Local Area Networks,” *IEEE Trans. On Consumer Electronics*, vol. 49, no. 4, pp. 861-871, November 2003.
[13] T. Shimizua, D. Shiraia, H. Takahashia, T. Murookaa, K. Obanaa, et al., “International Real-Time Streaming of 4K Digital Cinema,” *Future Generation Computer Systems*, vol. 22, no. 8, pp. 929-939, October 2006.
[14] B. Adamson, C. Bormann, M. Handley, and J. Macker, “Negative-acknowledgment (NACK)-Oriented Reliable Multicast (NORM) Protocol,” November 2004. Available: http://cs.itd.nrl.navy.mil/work/norm.
[15] Int. Standards Org./Int. Electrotech. Comm. (ISO/IEC), “Information technology—JPEG 2000 image coding system—Part 1: Core coding system,” *ISO/IEC 15 444-1:2000* and *ITU-T T.800*.
[16] Int. Standards Org./Int. Electrotech. Comm. (ISO/IEC), “JPEG 2000 Image Coding System—Part 11: Wireless,” *ISO/IEC WD2.0 15 444-11*, December 2003.
[17] Int. Standards Org./Int. Electrotech. Comm. (ISO/IEC), “JPEG 2000 Image Coding System—Part 8: Secure JPEG 2000,” *ISO/IEC WD2.0 15 444-8*, December 2003.
[18] “Advanced Video Coding for Generic Audiovisual Services,” *ITU-T Rec. H.264, ISO/IEC 14496-10*, March. 2005.
[19] “Information technology – Generic coding of moving pictures and associated audio information: Systems,” *ISO/IEC 13818-1*, 2005.
[20] SMPTE 377M-2004 Television – Material Exchange Format (MXF) – File Format Specification, *SMPTE*, 2004.
[21] Int. Standards Org./Int. Electrotech. Comm. (ISO/IEC), “JPEG 2000 Image Coding System—Part 3: Motion JPEG 2000,” *ISO/IEC WD2.0 15 444-3*, December 2003.
[22] CIE (1931). Commission Internationale de l'Eclairage Proceedings. Cambridge University Press, Cambridge.
[23] B. Adamson and J. Macker, “Quantitative prediction of NACK-oriented reliable multicast (NORM) feedback,” Available: http://pf.itd.nrl.navy.mil/norm.
[24] I. S. Reed and G. Solomon, “Polynomial Codes Over Certain Finite Fields,” *Journal of the Society for Industrial and Applied Mathematics*, Vol. 8, No. 2, pp. 300-304, June 1960.
[25] A. Rao, R. Lanphier, and H. Schulzrinne, “Real Time Streaming Protocol (RTSP),” *RFC 2326*, April 1998.
[26] S. Futemma, A. Leung, and E. Itakura, “RTP Payload Format for JPEG 2000 Video Streams,” *draft-ietf-avt-rtp-jpeg2000-10*, February 2006.
[27] High definition test sequences. Available: ftp://ftp.ldv.e-technik.tu-muenchen.de/pub/test_sequences.

Paolo Micanti was born in Foligno, Italy, in 1981. He graduated in telecommunications engineering at the University of Perugia in 2003, and he received the master degree in informatics and telecommunications engineering in 2006.

He is currently studying for his doctorate degree in telecommunications at the Department of Electronic and Information Engineering (DIEI), University of Perugia.

His main research interests are in the field of multimedia and image processing, and in satellite communications.

Giuseppe Baruffa was born in Perugia, Italy, in 1970. He received his *laurea* degree in electronic engineering from University of Perugia, Italy, in 1996, and his doctorate degree in telecommunications also from the University of Perugia in 2001. Since 2005, he has been an assistant professor at the Department of Electronic and Information Engineering (DIEI), University of Perugia.

His main research interests are in the field of digital broadcasting techniques (DVB), frequency assignment algorithms in nonlinear environments, joint source/channel coding for video over wireless networks.

Fabrizio Frescura was born in Perugia, Italy, in 1966. He received the *laurea* degree in electronic engineering from the University of Perugia in 1992 and the doctorate degree in 1997. He has been involved in personal communications, OFDM technique applied to Wireless LANs, digital audio and video broadcasting, and DSP algorithms and architectures for digital modems.

He is currently assistant professor at the Department of Electronic and Information Engineering (DIEI), University of Perugia, co-chair of the JPWL AHG, and co-editor of the Digital Cinema AHG in the ISO/IEC JTC1 SC29/WG1—JPEG Standards Committee. He is also one of the coordinators of the EDCINE IST research project.

His current research activity is focused towards wireless transmission of multimedia, DSP architectures for multimedia processing, and digital cinema.