PERFORMANCE OPTIMIZATION FOR MULTIMEDIA TRANSMISSION IN WIRELESS HOME NETWORKS

D. Melpignano, G. Convertino, A. Vitali,
AST, STMicroelectronics, Agrate, Italy
P. Bucciol,
IEIIT-CNR, Politecnico di Torino, Torino, Italy
A. Servetti, J.C. De Martin
Dipartimento di Automatica e Informatica, Politecnico di Torino, Torino, Italy

ABSTRACT

This paper describes a network adaptive real-time demonstrator for converged applications (audio, video, voice, and data) on an IEEE802.11g Wireless Home Network. Video transmission quality is optimised by dynamically adapting the source video bit-rate to a real-time estimate of the available bandwidth on the wireless network and by introducing data redundancy to recover packet losses (Forward Error Correction). Video adaptation is done by DCT-domain video transcoding algorithms performed in real-time on a digital signal processor.
Voice over Internet Protocol (VoIP) services are offered managing the coexistence of 802.11g terminals and Bluetooth headsets. Audio time-scale modification and adaptive playout algorithms enable robust and high quality interactive voice communications minimizing the effect of packet losses and jitter typical of wireless scenarios. All devices can share and remotely control content via Universal Plug and Play (UPnP).

1. INTRODUCTION

Improvements in wireless technologies are enabling scenarios where low-cost home devices are connected together. Content is easily accessed regardless of the physical place where it is stored. This will result in a large amount of data transmitted around the home among an increasing number of devices that share a common wireless medium.
In this scenario key requirements include: 1) ease-of-use, to hide all the installation, configuration and interoperability issues; 2) end-user Quality of Service provision.
The first need has been successfully addressed by company consortia that develop protocols to share multimedia content and to ease network setup. Universal Plug and Play Forum [1] and Digital Living Network Alliance [2] are becoming the de facto standards to develop home networking solutions.
Instead, QoS provision is still the most challenging problem in wireless multimedia distribution. Even if latest WLAN solutions offer high bandwidth, there are still problems related to interference, multipath fading and mobility that result in high (and variable) bit error rate and in variable bandwidth. New WLAN standards, such as IEEE 802.11e and the upcoming IEEE 802.11n, try to mitigate the problem providing additional services that help introducing QoS support, but this solution is not sufficient in situations with highly time-varying conditions. Moreover it will take some time before 802.11e/n devices will appear on the market.
In order to reach the goal of providing high level final user experience it is necessary to design multimedia applications that can cope with wireless transmission impairments. The main problems that applications should consider are:

- data losses. Real time/multimedia communication is usually based on unreliable transport protocol, like UDP, and the delay introduced by retransmissions of lost packets could be unacceptable for multimedia applications. UDP over 802.11 MAC can cause packet losses at the receiver, because MAC implementations discard the whole packets in presence of errors. The 802.11 MAC implements an ARQ mechanism (automatic repeat request) in presence of frame errors, but the number of retries is limited and hence there is not guarantee of correct frame delivery.
- bandwidth fluctuation, the presence of packet delivery errors causes the available wireless bandwidth to change over time. This is a consequence of retransmissions at MAC level as well PHY data rate adaptation scheme, usually implemented in WLAN devices to improve transmission robustness.

Many techniques have been proposed to solve these problems, an overview is provided in [3]. In this paper we do not provide an exhaustive overview, but we concentrate on the ones we consider more relevant and that we implemented and tested.
The rest of the document is organized as follows: in the next section a wireless home network (WHN) demonstrator is described, which applies optimisation algorithms discussed in section 3. In section 3.5 a way to co-ordinate such optimisation algorithms through an intelligent module called Cross-Layer-Controller (CLC) is provided. Conclusions are drawn in section 0.

2. DEMONSTRATOR

In Figure 1 the setup of a WHN demonstrator is depicted, where some of the algorithms further described in Section 3 were tested.

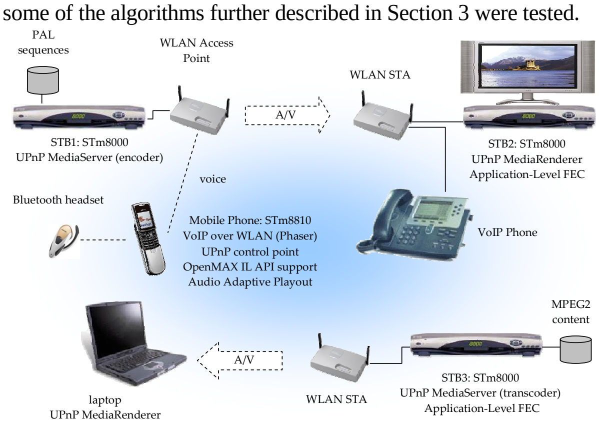

Figure 1: Demonstrator scenario.

Three Set-Top Boxes (based on the STm8000 programmable System-on-Chip) share audiovisual content through an IEEE 802.11g Wireless LAN using the UPnP protocol for signalling. As bandwidth available for A/V streaming varies with time, Dynamic Rate Shaper (DRS) transcoding is applied in the streaming server controlled by a CLC module. DRS parameters are set by CLC according to the buffer occupancy level behaviour of WLAN transmission queues. The server is able to support concurrent multiple multimedia stream delivery (one of which transcoded) and to dynamically insert an application-level Forward Error Correction (FEC) stream to protect individual multimedia flows. The traffic generated by two streaming servers is such that the WHN may be close to congestion.

In the same environment, a Nomadik™ STn8810 based mobile phone runs a Voice-over-IP application using the low-power STLC4370 WLAN chipset and sending voice to a Bluetooth headset, via STLC2500 module. Bluetooth and WLAN transceivers are coordinated through a dedicated hardware interface for operation on single-antenna low-cost devices. Audio Adaptive Playout (AAP) is used to improve robustness of the VoIP call.

The mobile phone also features a UPnP control point that allows managing other devices in the home network. For example, the mobile phone may select the server, the content and renderer for an A/V streaming session in the WHN.

3. ALGORITHMS

3.1 Video Transcoding

An efficient way to face bandwidth fluctuation is to adapt the encoding multimedia data rate according to the estimated available bandwidth. There are several ways in which the data rate of an encoded data stream can be reduced, depending on the way the stream is encoded. The solution we introduce in this section is a transcoder able to dynamically change the data rate of an encoded video stream. Our DRS transcoder works on MPEG-2 data streams (MPEG-2 to MPEG-2) and has the following features:

- **Bitrate change**: it allows decreasing the incoming bitrate of a compressed stream by requantizing in the DCT domain the coefficients to achieve a defined output rate.
- **Frame rate change**: it allows reducing the incoming frame rate of a compressed stream by dropping B frames. This is done in the compressed domain by header parsing, and just dropping the bits representing B frames selected for decimation.
- **Frame size change**: it allows reducing to 1:4 the image resolution (by reducing to 1:2 both horizontal and vertical resolutions). The prediction error downsampling is always done into the DCT domain, in order to reduce the requirements in term of memory buffers. The bitstream reshaping can be done both in DCT or pixel domains. In particular, the prediction error downsampling works by cropping the lower frequency 4x4 coefficients of each block of a macroblock, and fusing them together into a single new 8x8 DCT block.
- An automatic algorithm is able to **optimally select** among quantization increase, frame rate or size reduction, given a target bitrate, in order to achieve the best possible visual quality of the transcoded stream

The DRS has an advantage in comparison to the classical decoding/encoding chain also in terms of quality. Since motion vectors are computed from the original encoder on high quality images, the effect of errors and noise that can reduce performances of motion estimation on decoded images is eliminated. The transcoder has advantages in term of memory size and bandwidth (no frame buffers at all), in term of computation (motion estimation and compensation are not necessary) but also in term of quality of the final result. In Figure 2 it is possible to see the PSNR measure for luma channel of the full chain (coding – decoding – re-encoding [Co-Dec-Rec]) in blue and the PSNR for our transcoder in red. The sequence under test is the well known "Mobile&Calendar" transcoded from 7 to 4 Mbit/s. The red line is above or equal to the blue line but our DRS is an order of magnitude less complex compared to decoding and re-encoding. The green line represents DRS performance with dithering algorithm turned on. This special quantizer gives us a benefit in terms of quality as the green line is the PSNR for luma & chroma when the sequence "calendar" is transcoded using this option. The green line is often above the red line, classical linear re-quantizer is used.

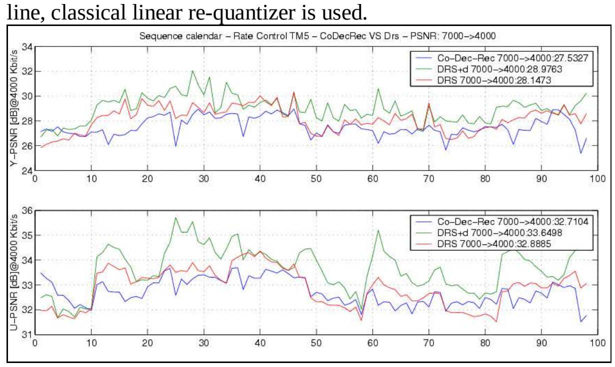

Figure 2: DRS performance.

3.2 Application Level FEC

FEC schemes have been proposed by many researchers to make applications more resilient to packet losses [4]. FEC techniques rely on the transmission of redundant information from which lost packets can be recovered; this approach reduces the packet loss recovery time compared to ARQ schemes.

The FEC encoder works on the sender side and generates a new block of n packets from a block of k data (or *source*) packets,, where (n-k) FEC redundant (or *parity*) packets are transmitted. On the receiver side a FEC decoder recovers lost data packets using both received data and parity packets.

In our implementation, we have integrated and we can alternatively use two FEC codecs: Reed-Solomon (RS) and Low Density Parity Check (LDPC). Some of the most interesting features of FEC schemes are the following:

- FEC encoding provides great advantages in terms of protection of data to losses over the network: the correct reception of at least k packets (data or FEC) of the n sent, is enough to reconstruct, at client side, all the k data packets (this is always true for RS codec, while LDPC may require the reception of more than k packets to recover k data packets).
- No delay is introduced in the encoding phase. Data packets are sent over the network, buffered and used by the FEC codec.
- FEC packets can be easily discarded by clients that do not support FEC decoding. For example using RTP encapsulation, FEC RTP packets can be recognized by the RTP payload type and can be sent in two ways: together with data packets or over a different connection as an enhancement layer. In both cases it is very simple to discard FEC packets, in the first case RTP client should discard the packet with a payload type that does not recognize, in the later the connection is not opened at all.

The aspects that must be carefully considered are:

- FEC decoding process introduces some delay when data losses occur. The delay must fit application requirements.
- The FEC encoding/decoding introduces some overhead in computation that must be kept as low as possible. It is very im-

portant to use efficient codecs, LDPC codecs are very satisfactory from this point of view.

- FEC redundancy can waste network resources if it is not tuned according to network conditions

The best way to introduce FEC without wasting network resources is to dynamically introduce redundancy according to the current link conditions. We propose a decision module that computes the n and k parameters according to: the maximum delay the application can tolerate and the wireless packet error rate (PER), see section 3.5. FEC recovery results using the described approach are shown in Figure 3.

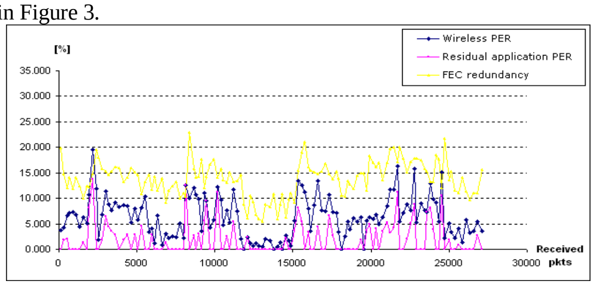

Figure 3: FEC recovery example: the loss rate after recovery (Residual application PER, violet line) is always less than the actual loss rate experienced on the WLAN (blue line), thanks to FEC redundancy added to the stream (yellow line). FEC redundancy is dynamically tuned according to the past history of the WLAN loss rate, to save network bandwidth.

3.3 Wireless perceptual ARQ

The standard IEEE802.11 WLAN architecture includes an ARQ mechanism which operates at the MAC level by retransmitting corrupted packets regardless of their relevance to a specific application. While this mechanism is satisfactory for homogeneous traffic, such as data traffic, multimedia streams present packets with specific characteristics: highly non-uniform perceptual importance and strong time sensitivity. One or both aspects are usually considered by most multimedia specific ARQ techniques.

The Soft ARQ proposal, for instance, uses the time deadline of each packet to avoid retransmitting late data that would not be useful at the decoder, thus saving bandwidth. Variants of the Soft ARQ technique have been developed for layered coding [5]. Other techniques exploit the different perceptual importance of the syntax elements contained in a compressed multimedia bit stream by means of a prioritization mechanism [6][7].

We developed a perceptual ARQ scheme (WP-ARQ) [8], implemented at the application level, which exploits information about the perceptual and the temporal importance of each packet. In our proposal, a set of retransmission opportunities is determined to allow the algorithm to re-transmit unacknowledged packets according to their priority. Each packet's priority is computed by combining perceptual importance (evaluated using the analysis-by-synthesis technique [9]) and maximum delay constraint. Compared to the standard 802.11 MAC-layer ARQ scheme, the proposed technique delivers higher perceptual quality because it retransmits only the most perceptually important packets.

The impact of a number of parameters on the WP-ARQ algorithm has been considered. The amount of bandwidth allocated for retransmission purposes, for instance, influences the performance of the algorithm deeply. Better PSNR results can be achieved when more bandwidth is available; however, the gain is less pronounced when the maximum available retransmission bandwidth value is already high.

We also introduced a parameter to weight the relative importance between perceptual and temporal information, thus prioritizing either the perceptual importance or the temporal importance of each packet. Study and simulations showed that, when the network load is low, the packets with more temporal importance should be privileged. When the network is congested, more perceptually important packets should be privileged. However, the optimal operating point depends on the characteristics of the sequence. For a static sequence the number of packets with a high perceptual importance is very limited, therefore the best strategy is always to privilege the perceptually important packets.

Results showed that in low congestion scenarios the pro-posed method has limited gain over the standard 802.11 ARQ mechanism, because the MAC level ARQ is more aggressive.

On the other hand, when congestion increases, our proposed method consistently outperforms the standard ARQ mechanism, delivering PSNR gains up to 10 dB, as well as very low transmission delay and limited impact on concurrent traffic.

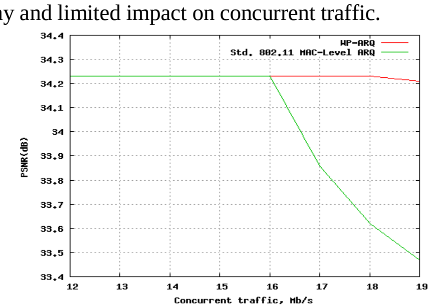

Figure 3: Performance comparison, in terms of PSNR, between standard 802.11 ARQ (green) and WP-ARQ (red) over a 36Mb/s Wireless LAN.

3.4 Multiple description

Another way to increase error resilience is to use Multiple Description coding (MD) [13][14]. The goal of MD is to create several independent descriptions which are independent of each other. Descriptions can have the same importance (as for balanced MD) or they can have different importance (as for unbalanced MD). They can be decoded independently or jointly. The more descriptions decoded, the higher the output quality. Error resilience comes from the fact that it is unlikely to have the same portion of data corrupted in every description.

The simplest way to create multiple descriptions by using an existing video codec is to work in the data domain [15][16][17]. Descriptions are created in a pre-processing stage; then, they can be independently encoded, transmitted and decoded; finally successfully received and decoded descriptions are merged in a post-processing stage. As an example, two descriptions can be created by separating odd and even lines. Variable bandwidth/throughput can be easily managed by transmitting a suitable number of descriptions. No transcoding is needed to match the channel capacity. MD can also exploit path diversity. Of course, coding efficiency is somewhat reduced because pixels are less correlated and bitstream syntax is replicated in each description. However, any technique that introduces scalability or error resilience does reduce coding efficiency [18]. Main MD applications are summarized below:

- Divide-et-impera approach for HDTV distribution: HDTV sequences can be split into four SDTV descriptions; no custom high-bandwidth h/w is required.
- Adaptation to varying bandwidth: the base station can simply drop descriptions; more users can be easily served, and no transcoding process is needed.
- Adaptation to low resolution/memory/power: mobiles decode as many descriptions as they can, based on their display size, available memory, processor speed, and battery level.

- Easy picture-in-picture: with the classical solution, a second full-decoding is needed plus downsizing; with MD, it is sufficient to decode one description and paste it on the display.

Figure 4: Comparison of MD (4 descriptions) and the usual single description transmitted over an error prone channel, before and after error concealment. Same aggregate bitrate, same average packet size.

ARQ and FEC can be used in conjunction with MD. The protection level of a given description should match its importance, a technique commonly known as Unequal Error Protection (UEP) and usually applied to Layered Coding. UEP can be used even in case of equally important descriptions (balanced MD). In fact, armoring only one description may be more effective than trying to protect all descriptions.

MD has been benchmarked against Reed-Solomon application layer FEC [18]. Results indicate that FEC is preferable when its correction capability is not exceed, hence for low packet loss rates. On the opposite MD is preferable for medium to high packet loss rates. In the latter case MD yields a higher PSNR with lower variance.

3.5 Cross-Layer Controller

The optimisation techniques presented in the previous sections usually have an optimal operating point, which depends on the wireless network state. For example, using FEC is an unnecessary waste of bandwidth when the link is good. Also, when WP-ARQ outperforms MAC layer ARQ, it may be worth reducing the maximum number of MAC retransmissions to reduce offered traffic.

Cross-layer optimisation refers to application-specific tuning of parameters at different layers of the protocol stack. For multimedia transmission over wireless the “radio with knobs” ([20]) concept can be used. In particular, a software entity is introduced, named Cross-Layer Controller (CLC), which collects measurements of streaming environment and consequently sets parameters of the WLAN MAC and PHY layers, and decide the optimal bitrate for video and FEC data (see Figure 5).

The implemented algorithm is quite simple and does perform very light processing (no relevant computation is required): as the measured size of the transmission queue overcomes a given threshold, the CLC selects a new bitrate value for the video stream, among a discrete interval of preset values, corresponding to different video quality levels. The selected value, increased by the amount of FEC data, must not overcome the measured available bandwidth. The amount of FEC data is computed as the measured PER plus a fixed margin (to face possible increases of losses in the near future).

A key enabler for CLC - from an implementation perspective - is the Application Programmer’s Interface to collect wireless link statistics and set parameters at different layers of the stack. In this field, a Universal Link Layer API to control wireless links is being investigated [21], which provides abstraction of different link-layer technologies, with statistics querying and asynchronous notification capabilities. It is worth noting that CLC is only trying to apply adaptation on a long-term time scale.

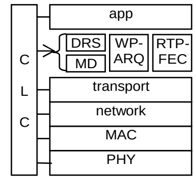

Figure 5: The position of the CLC in the protocol stack.

3.6 Robust decoding – Error concealment

It is often useful to jointly optimize the parameters of the source and channel encoders. In the case of multimedia communications this means exploiting the error resiliency that can be embedded in compressed bitstreams rather than blindly using complex ARQ/FEC schemes.

For the case of hybrid video coders like MPEG-2 or H.264 it is possible to increase the error resiliency by using one or more of the following techniques: using more frequent intra pictures or a suitable macroblock intra refresh policy to reset the motion prediction loop and stop error propagation; more slices per picture to reset differential motion vector and DC transform coefficient coding; flexible macroblock order (FMO) and/or asynchronous slice order (ASO) so that a burst of errors result in scattered losses in the picture, making it easier to conceal losses; using interleaved multiframe prediction policies to reduce dependency among pictures; encoding concealment motion vectors or redundant slices (with a coarser quantization step) used as a replacement to conceal losses; using an error resilient entropy coding scheme (EREC), reversible variable length codes (RVLS) or more sync markers to limit the effect of errors on the compressed bitstream.

A robust decoder should be able to detect and skip bit-stream errors exploiting the knowledge of the syntax [19]. For the case of H.264 decoders, every parameter in the bit-stream has a well defined range, every violation indicates an error. As an example, the frame number is a crucial parameter for correct decoding, display process and frame loss detection; it can be checked against the expected range. As another example, a certain number of parameters is in-variant in slice headers. Therefore errors in slice headers are not only detectable but they can even be corrected.

A robust decoder should also be able to conceal missing data exploiting the spatial/temporal correlation of the video source [22]. Standard techniques are: maximally smooth spatial interpolation; motion compensated copy from temporally adjacent frames where motion vectors are computed based on neighboring data (motion vectors or pixels); an adaptive selection of these methods based on motion content and on scene change detection: static or low-motion frames are concealed using the temporal algorithm, high-motion frames or scene changes are concealed using the spatial algorithm. The loss of one or more whole frames can also be concealed using sophisticated algorithms [23].

3.7 Adaptive Audio Playout

One of the major challenges for reliable and high quality interactive VoIP services over an 802.11g wireless network is the high variability of the delays (jitter) experienced by audio frames sent through the wireless network and the backbone. To protect the system against random delays that may cause silence gaps and playout errors , a buffer with limited capacity is used at the receiver

to smooth out the delay variations of received packets. The robustness of the system is thus increased at the expenses of a higher end-to-end delay and reduced interactivity.
To optimise the trade-off between robustness and interactivity we provided this demonstrator with an adaptive audio playout (AAP) algorithm that adaptively adjusts the playout buffering in order to keep the delay as small as possible, while at the same time maintaining robustness against losses due to the arrival of packets after their playout time threshold. To improve the performance of the algorithm playout adjustments are not only performed during silence periods (as in [10]), but also during active speech periods. The continuous output of high quality audio is assured by scaling the audio frames using the WSOLA time-scale modification technique that can change the frame duration without altering the frequency content [11].

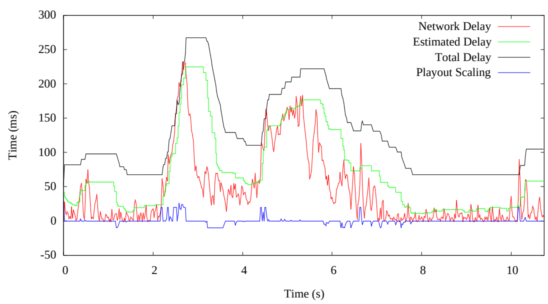

Figure 7: Adaptive audio playout example.

Dynamic adaptation of the playout delay is achieved by varying the playout speed of audio frames depending on the channel conditions [12]. That is, playing the media slowly when the buffer occupancy is below a desired level and playing faster then real-time when the channel conditions are good in order to eliminate excessive latency accumulated during bad channel periods. As shown in Figure 7, for each frame, the adaptive playout scheduler estimates the expected delay of the next frame (green line). If required, the buffering delay may be modified by scaling the duration of the current frame by a given factor (blue), so that the total delay experienced by the audio packets (black) may closely follow the network delay (red).
Besides its use to enable adaptive playout buffering, we have also employed the audio time-scaling technique as a packet loss concealment scheme. Instead of repeating the last received frame, improved audio quality is obtained by stretching the neighbouring frames to fill the gap corresponding to lost frames.

4. CONCLUSIONS

In this paper, we have analysed how multimedia sessions in a wireless home network can be managed so that typical impairments are mitigated and the resulting user experience is maximised. Out of the possible algorithms to improve robustness against packet losses and jitter, we have tested a few of them in a real-time demonstrator, where measurements could be collected in realistic situations.
In our experience matching computational cost of optimisation algorithms with benefits in terms of perceptual quality improvement is a tough task, which requires extensive experimentation in real-world scenarios.
A cross-layer approach based on the joint optimisation of parameters at different layers of the protocol stack (from the application down to the physical layer) is already viable with today's technology and provides measurable advantages. Our prototype uses a CLC software module to activate the proper optimisation algorithms in a coordinated way, depending on application needs.

REFERENCES

[1] UPnP forum home page: http://www.upnp.org
[2] DLNA home page: http://www.dlna.org
[3] Y. Wang, et al., "Error resilient video coding techniques", IEEE Signal Processing Mag. vol. 17, no. 4, pp.~61-82, July 2000
[4] C. Perkins, et al. "A Survey of Packet Loss Recovery Techniques for Streaming Media" IEEE Network Magazine, pp. 40--48, September/October 1998.
[5] M. Podolsky, et al. "Soft ARQ for layered streaming media," in Tech. Rep. UCB/CSD-98-1024, University of California, Computer Science Division, Berkeley, November 1998.
[6] Y. Shan and A. Zakhor, "Cross layer techniques for adaptive video streaming over wireless networks," in Proc. IEEE Int. Conf. on Multimedia & Expo, vol. 1, August 2002, pp. 277–280.
[7] S. H. Kang and A. Zakhor, "Packet scheduling algorithm for wireless video streaming," in Proc. Packet Video Workshop, Pittsburgh, PA, April 2002.
[8] P. Bucciol, E. Masala, J.C. De Martin, "Cross-Layer Perceptual ARQ for H.264 Video Streaming over 802.11 Wireless Networks," Proceedings of IEEE GLOBE-COM, Dallas, TX, USA, November-December 2004, vol. 5, pp. 3027-3031.
[9] E. Masala and J. D. Martin, "Analysis-by-synthesis distortion computation for rate-distortion opti-mized multimedia streaming," in Proc. 18 IEEE Int. Conf. on Multimedia & Expo, vol. 3, Baltimore, MD, July 2003, pp. 345–348.
[10] R. Ramjee, et al., "Adaptive playout mechanisms for packetized audio application in wide-area networks," in Proc. INFOCOM, pp. 680-688, 1994
[11] W. Verhelst, "Overlap-Add Methods for Time-Scaling of Speech," Speech Communication, vol. 30, no. 4, pp. 207-221, 2000
[12] Y.J. Liang, N. Farber, and B. Girod, "Adaptive playout scheduling using time-scale modification in packet voice communications," in Proc. ICASSP, vol. 3, pp. 1445-1448, May 2001.
[13] John G. Apostolopoulos "Video Streaming: Concepts, Algorithms and Systems" HP Laboratories, report HPL-2002-260, September 2002.
[14] V.K. Goyal. Multiple description coding: compression meets the network. IEEE Signal Proc. Mag. 18 (5) (Sep-tember 2001) 74–93.
[15] S. Shirani, M. Gallant, F. Kossentini, "Multiple Description Image Coding Using Pre- and Post-Processing" in International TCC, Las Vegas, Nevada, USA, April 2000.
[16] R. Bernardini, et al."Polyphase Spatial Subsampling Multiple Description Coding of Video Streams with H264" in Proceedings of ICIP 2004, Singapore, pp. 3213-3216, October 2004.
[17] N. Franchi, et al. "Multiple Description Video Coding for Scalable and Robust Transmission over IP" IEEE Transactions on CSVT, vol. 15, no. 3, pp. 321-334, March 2005.
[18] R. Bernardini, M. Durigon, R. Rinaldo, A. Vitali, "Comparison between multiple description and single description video coding with forward error correction", MMSP 2005.
[19] G. Gennari, et al. "Slice header reconstruction for H.264/AVC robust decoders", MMSP 2005.
[20] M. Zorzi, R.R. Rao, "Perspectives on the Impact of Error Statistics on Protocols for Wireless Networks", IEEE Personal Communications, vol. , n. , pp.32-40, October 1999.
[21] T. Farnham et al., "Towards Open and Unified Link Layer API", IST Mobile Summit 2005.
[22] G. Gennari, G. A. Mian, L. Celetto, "H.264 robust to transmission errors", EUSIPCO 2004.
[23] E. Quacchio, E.Magli, G. Olmo, P. Baccichet, A. Chimienti, "Enhancing whole frame error concealment with an intra motion vector estimator in H.264/AVC" ICASSP 2005.