Deliverable D2e

Impact of semantic dependencies in MPEG-4 bitstream on timing constraints

Consortium Confidential

Information Society
Technologies

Project Number	:	IST-004042
Project Title	:	BETSY
Deliverable Type	:	Report

Deliverable Number	:	D2e
Title of Deliverable	:	Impact of semantic dependencies in MPEG-4 bitstream on timing constraints
Nature of Deliverable	:	Report, Restricted
Internal Document Number	:	v 2.0
Contractual Delivery Date	:	28 August 2006
Actual Delivery Date	:	28 August 2006
Contributing WPs	:	WP2
Author(s)	:	C. Blanch (IMEC) D. Isovic (MDH) G. Fohler (MDH) P.v.d. Stok (Philips)

version 2 after second review

Abstract

This deliverable analyzes the impact of the MPEG-4 semantics on the timing constraints of functional components identified in D2a, such as encoding, decoding and transmission functions. This way, the impact of the percentage of Inter/Intra coded information on encoding/decoding latencies is analyzed on several sequences, as this parameter can be tuned to satisfy timing constraints and make tradeoffs between timeliness, resource usage and quality. Finally, the mechanisms applied when the decoding timing constraints cannot be met are also presented.

Keyword list

MPEG-4 semantics, timing constraints, Frame latency, Intra coded, Inter coded

Table of Contents

1 Introduction

This deliverable analyzes the impact of the semantics of the MPEG-4 codec on the timing constraints on entities present in deliverable D2a [1]. The following sections introduce the impact of the semantics on individual component latencies that contribute to the end-to-end latency.

1.1 End-to-end Timing Constraints

Video applications are subject to throughput and real-time constraints. Different components of the end-to-end system, such as encoder, decoder, buffers and network components have an impact on the end-to-end latency that needs to be respected to avoid severe quality degradation.

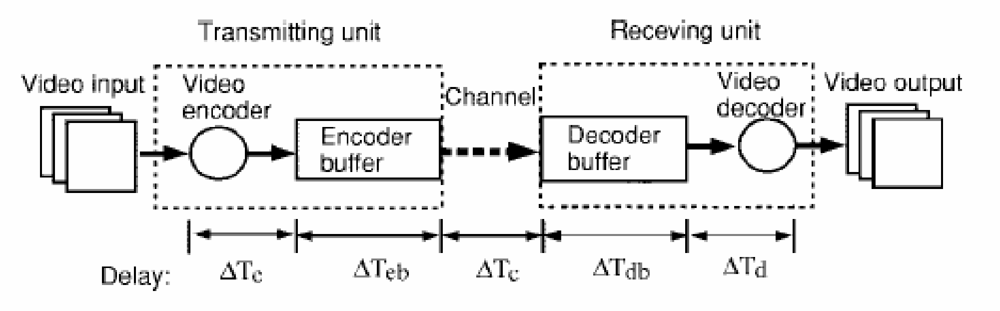

Figure 1.1: Delay components of a communication system

In typical video communications systems, the end-to-end delay each frame experiences (from the time it is obtained from the input video buffer to the time it is placed in the display video buffer) consists of several delay components [1]. The total delay experienced by each frame (see Figure 1.1) can be broken up into:

$$\begin{aligned}\Delta T = \Delta T_e \quad & \text{(Encoder delay)} \\ + \Delta T_{eb} \quad & \text{(Encoder buffer delay)} \\ + \Delta T_c \quad & \text{(Channel delay)} \\ + \Delta T_{db} \quad & \text{(Decoder buffer delay)} \\ + \Delta T_d \quad & \text{(Decoder delay)}.\end{aligned}$$

The semantics of MPEG-4 play an important role when analyzing both the indivual and end-to-end latency constraints. Missing the individual or global deadline of different kind of video information has a very different impact on the end-video quality. The following section introduces the impact of the semantic dependencies on the timing constraints.

2 Encoding Timing Constraints

The factors that determine the timing constraints for an MPEG video encoder are the following:

- The video sequence characteristics (degree of motion and detail): this will have an impact on the complexity of the encoding process and therefore on the capacity to meet the timing constraints.
- Frame ordering and dependencies: there are dependencies between P and B frames. In a frame sequence such as I – P1 – B – P2, frames P1 and P2 need to be encoded prior to frame B as they are both used in the prediction of the B frame. Nevertheless, frame B needs to be displayed before P2. In the case of the MPEG-4 SP no B frames are used, therefore every frame can be encoded immediately after encoding of the previous frame (I or P).
- Camera rate constraints: the camera produces input data for the encoding process at a given rate. If the camera rate is too high for the encoder, the latter can skip some frames and encode only a subset of them (for example halving the frame rate).
- Timing constraints from other components in the communication system: as seen in Figure 1 the total delay experienced by each frame is composed of several delay components, one of them being the encoding delay. To meet the total delay constraint, an excess delay of one component (such as the network for example) could motivate the decrease in the delay of another component. This way, a network with low capacity (high transmission delay) or a slow decoder (high decoding delay) can pose harder constraints on the encoding delay that will need to be short to meet the overall delay.
- The MPEG-4 semantics have an impact as well on the encoding complexity, and therefore on the encoding latency. The following section describes the MPEG-4 semantics and its impact on the timing constraints.

2.1 MPEG-4 Semantics

The MPEG-4 standard [2] defines three types of frames, I, P and B. I-frames or intra frames are simply frames coded as still images. They contain absolute picture data and are self-contained, meaning that they require no additional information for decoding. I frames have only spatial redundancy providing the least compression among all frame types. Therefore they are not transmitted more frequently than necessary. The second kind of frames is P or predicted frames. They are forward predicted from the most recently reconstructed I or P frame, i.e., they contain a set of instructions to convert the previous picture into the current one. P frames are not self-contained, i.e., if the previous reference frame is lost, full decoding is impossible. On average, P frames require roughly half the data of an I frame. The third type is B or bi-directionally predicted frames. They use both forward and backward prediction, i.e., a B frame can be decoded from a previous I or P frame, and from a later I or P frame. They contain vectors describing where in an earlier or later pictures data should be taken from. They also contain transformation coefficients that provide the correction. B frames are never predicted from each other, only from I or P frames. As a consequence, no other frames depend on B frames. B frames require resource-intensive compression techniques but they also exhibit the highest compression ratio.

2.2 Impact of Semantic dependencies on Encoding Timing Constraints

On one hand, coding of Predictive information (P-frames) involves a computationally intensive process called Motion Estimation [3], which provides a higher compression degree than Intra prediction. On the other hand, coding Intra information (I-frames) can achieve less compression but it is also a less complex process. This way, a higher compression degree comes with an extra complexity cost and therefore the encoding latency of Predictive information is higher than the one of Intra information.

2.2.1 Use of Rate Control Mechanism

The MPEG-4 Visual standard requires each video frame or object to be processed in units of a macroblock. If the control parameters of a video encoder are kept constant (e.g. motion estimation search area, quantisation step size, etc.), then the number of coded bits produced for each macroblock will change depending on the content of the video frame, causing the bit rate of the encoder output (measured in bits per coded frame or bits per second of video) to vary. Typically, an encoder with constant parameters will produce more bits when there is high motion and/or detail in the input sequence and fewer bits when there is low motion and/or detail. Additionally, the degree in which we use Intra frame prediction (I frames or I IntraMBs) or Inter frame prediction (P frames or P MBs) has a major impact on the output bit rate produced. Inter coded information achieves a higher compression efficiency and yields therefore lower output bit rate for the same coding quality (quantization step size).

This variation in bit rate can be a problem for many practical delivery and storage mechanisms. For example, a constant bit rate channel cannot transport a variable-bit rate data stream. A packet-switched network can support varying throughput rate but the mean throughput at any point in time is limited by factors such as link rates and congestion. In these cases it is necessary to adapt or control the bit rate produced by a video encoder to match the available bit rate of the transmission mechanisms.

The Rate Control mechanism involves modifying the encoding parameters in order to maintain a target output bitrate. The most obvious parameter to vary is the quantiser parameter or step size (QP) since increasing QP reduces coded bitrate (at the expense of lower decoded quality) and viceversa. A common approach to rate control is to modify QP during encoding in order to (a) maintain a target bitrate (or mean bitrate) and (b) minimise distortion in the decoded sequence.

The use of Rate Control mechanism [1] typically smoothens down the bit rate variation although instantaneous rate variations of even a factor 2 or 3 may occur. In the following sections we study the impact of the MPEG-4 semantics on the encoding timing constraints. This is done by analyzing the impact of the degree of Intra and Inter information during the coding process. As the insertion of Intra information generally increases the output bit rate, the analysis is done using the Rate Control mechanism so that the comparisons are done for the same target bit rate. As the output rate of the encoder remains the same, we don't have to consider the impact of a higher overall encoder output rate on the network timing constraints.

2.2.2 Gradual Intra Refresh versus Use of Intra frames

According to the standard, every video Macroblock (16x16 block within a video frame) needs to be Intra updated every 132 frames [4]. This can be achieved by means of introducing an Intra frame every 132 frames. In practice, however video Intra information can be distributed along all P frames to refresh the video information without introducing Intra frames. This is done to avoid high peaks of the transmission rate caused by the insertion of an I frame. A more even distribution of the less compressed Intra information is made by introducing a certain degree of Intra coded Macroblocks in every P frame. This guarantees that after a certain number of frames all Macroblocks in a frame have been Intra updated.

When a frame or part of a video frame is lost during transmission, not only the quality of that frame is impaired but also the error propagates to following P frames, which use the lost information for prediction. This way, if frame P0 is lost, the decoder will use the concealed P0, frame for prediction of the next P1 frame, degrading the P1 frame quality, which in turn will affect the frame predicted from P1: P2 and so on.

This error propagation is stopped when next frames contain portions of Intra coded information, which are self contained and not predicted from previous frames. This can be done by forcing some Intra Macroblocks of the P frame to be Intra coded. The higher the percentage of Intra coded information the P frame contains, the sooner the error propagation is stopped and the errors corrected.

This gradual refreshment of Macroblocks has a twofold beneficial effect: it smoothens down the peak transmission rate and it increases the error resilience of the video sequence.

The periodical insertion of Intra frames creates peaks in the transmission but also provides error resilience by stopping the error propagation of previous frames. Intra frame insertion may be preferred when random access capabilities need to be provided.

The following sections analyze the impact of the amount of Intra coded information for both Intra Frame insertion approach and Gradual Intra Refreshment.

The analysis is done with a test bench. The test bench is composed of the widely known video sequences *Akiyo*, *Foreman,* and *Mobile and Calendar*. Akiyo is representative for the so-called ‘head and shoulders’ sequence, as it presents a news reporter with very little motion throughout the sequence and very low bit rate demands. The Foreman sequence has medium bit rate demands and medium degree of motion, which is non homogeneous along the sequence. Finally, Calendar and Mobile is a video sequence with high detailed texture, high degree of homogeneous motion and high bit rate demands.

2.2.2.1 Intra Frame Insertion

The following graphs analyze the impact of the amount of Intra coded information on the encoding latency when this is done by means of full Intra frame insertion. Time measurements are given for an MPEG-4 encoder [5] running on a Pentium 4 at 2.8 GHz. Note that the actual latency values are highly dependent on the platform where the encoder runs and on other implementation issues. Nevertheless, we can use these measurements on a PC to provide some relative speed up and express the impact of the MPEG-4 semantics. The encoder configuration also has an impact on the resulting latency numbers. The encoder in [5] uses an integer Discrete Cosine Transform (DCT) and Inverse DCT, integer precision for the motion vectors, and a search range of 16 pixels. No deblocking filter is implemented in its decoder.

The vertical axis of all figures in this deliverable represents the total encoding latency for several video sequences. In Figure 2.1 the values of Akiyo (blue), Foreman (red) and Mobile (black) are normalized to the highest encoding latency for each sequence. The horizontal axis represents the percentage of forced Intra Macroblocks in the sequence. This is done inserting full Intra frames periodically so that the Intra MBs concentrate in the Intra frames (all MBs are Intra coded) and not in the P frames (MBs are predictively coded). Table 2.1 shows the relation between the Intra frame period and the resulting percentage of Intra MBs for the whole sequence.

Intra Frame Period	20	9	6	5	3
% Intra MB	5.5%	11%	16.6%	20%	33.3%

Table 2.1: Intra Frame Period vs % Intra MB

A higher number of Intra Frames inserted (corresponding to a smaller Intra Frame Period) corresponds to a higher percentage of Intra coded MBs over the sequence. We can see in Figure 2.1 that the higher the percentage of Intra MBs the lower the encoding latency is for sequences with medium to high motion like Foreman and Mobile, achieving for 33% of Intra MBs speed ups of 15% for Intra Frame Insertion. On the contrary, for the low motion Akiyo sequence at 150 Kbps the introduction of Intra frames increases the latency by 5%, as the low motion allows an easy estimation from previous frames in the P frames.

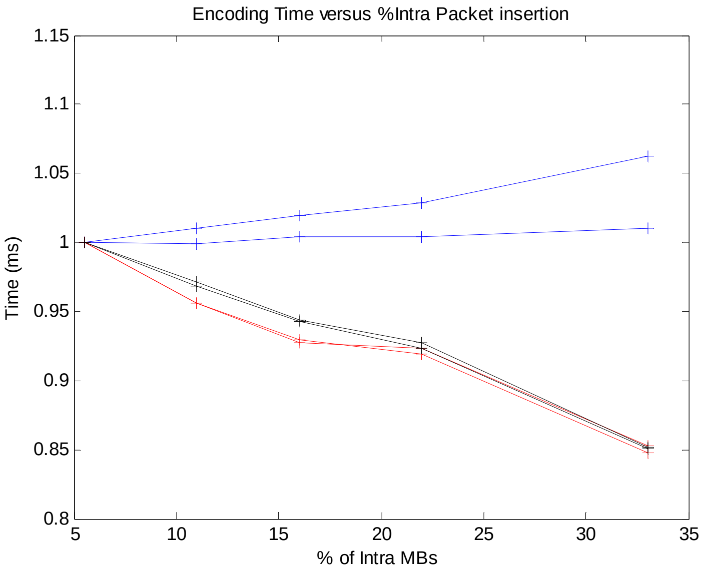

Figure 2.1: Encoding latency versus Intra Frame insertion

Table 2.2 shows the variability in the total encoding latency between the considered sequences. We can see how complex sequences with a high degree of motion (Mobile) require around factor 4 higher encoding latency than low motion sequences like Akiyo. The target rate of the sequence influences the quantization parameter (QP) used, and this also has a noticeable impact on the latency requirements.

	Akiyo (150Kbps)	Akiyo (400Kbps)	Foreman (1 Mbps)	Foreman (1.5 Mbps)	Mobile (2Mbps)	Mobile (6Mbps)
Total Latency	2.43 s	3.35 s	5.98 s	7.21 s	8.16 s	9.23 s

Table 2.2: Total encoding latency for 5.5% Intra MB

It is also interesting to observe the instantaneous encoding latency or frame encoding latencies. In Figure 2.2 and Figure 2.3 the encoding latency for the Foreman sequence is shown, we can identify as Intra frames the frames with lowest encoding latency (almost half the encoding latency of P frames). In Figure 2.3 the latency oscillations become much more noticeable as a 33% of Intra MB corresponds to an Intra frame inserted every 3 frames (IPPIPP scheme).

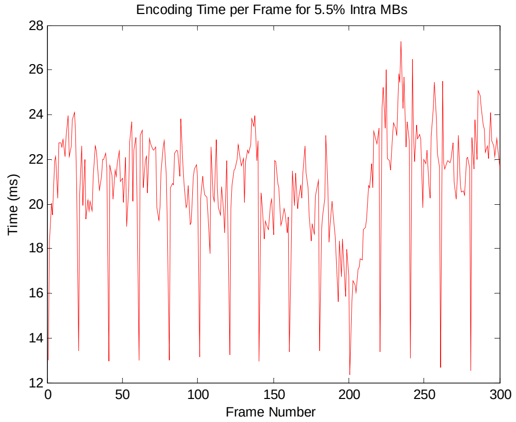

Figure 2.2: Frame Encoding Latency for Intra Period 20, Foreman at 1Mbps

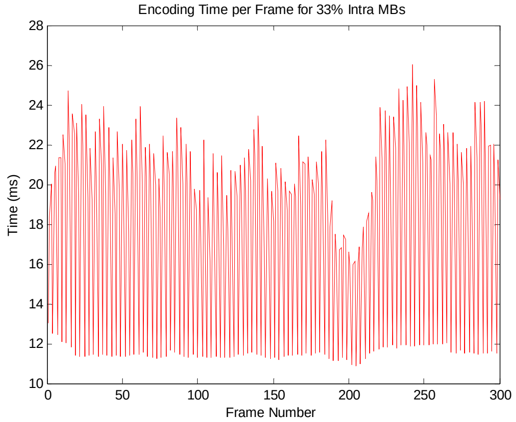

Figure 2.3: Encoding latency for Intra period 3, Foreman 1 Mbit/s

From the figures and from Table 2.3 we can extract that the inter frame latency variations are much stronger for a higher number of Intra MBs (or Intra frames in particular) inserted. We can see that the average encoding latency diminishes by inserting more Intra frames but on the other hand, the latency standard deviation highly increases. This can pose more difficulties for the rest of components in the end-to-end chain, as the output pattern of the encoder is highly variable (though predictable if the Intra Frame period is known).

% Intra MB	Average Latency	Standard Deviation	Max Latency	Min Latency	Max/Min Ratio
5.5%	20.86 ms	2.63 ms	27.26 ms	12.35 ms	2.2
33%	17.66 ms	4.68 ms	26.02 ms	10.84 ms	2.4

Table 2.3: Latency statistics for Foreman

The same analysis is done for the Mobile sequence in Figures 2.4 and 2.5 and Table 2.4, where the same conclusions as for the Foreman sequence can be extracted. However, the analysis of the Akiyo sequence (Figure 2.6, Figure 2.7, Table 2.5) differs from the previous sequences. While in Figure 2.4 (Mobile) the drops in the encoding latency correspond to Intra frames, in Figure 2.6 for Akiyo the Intra frames cause latency peaks, meaning that for a low motion sequence as Akiyo, the encoding of predictive information can be less complex than the encoding of Intra information, in which the high temporal correlation is not exploited. This explains that the encoding latency slightly increases from 5.5 % to 33% of Intra MB. As for the latency variability between frames, as in the case of Foreman and Mobile, the latency variability is increased with 33%, as we can see in the value of the standard deviation.

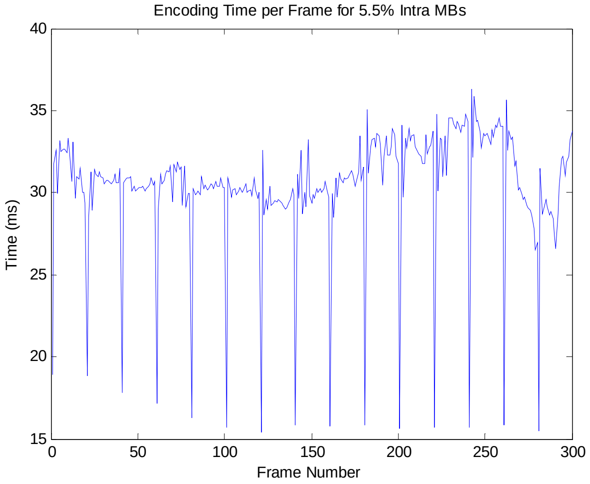

Figure 2.4: Frame Encoding Latency for Intra Period 20, Mobile at 6 Mbps.

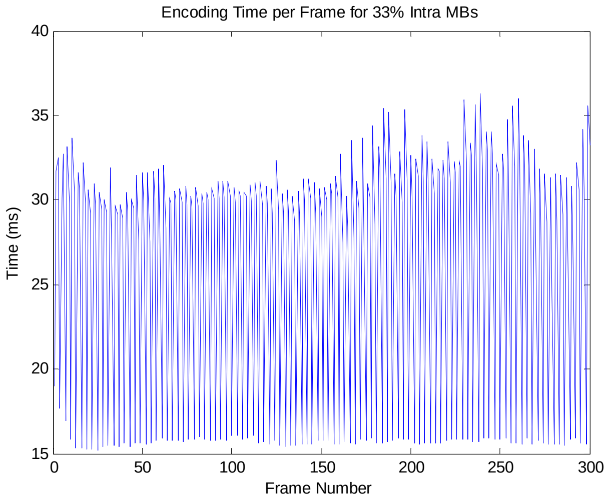

Figure 2.5: Encoding latency for Intra period 3, Mobile at 6 Mbps.

% Intra MB	Average Latency	Standard Deviation	Max Latency	Min Latency	Max/Min Ratio
5.5%	30.48 ms	3.67 ms	36.30 ms	15.38 ms	2.36
33%	25.92 ms	7.37 ms	36.29 ms	15.16 ms	2.39

Table 2.4 Latency statistics, Mobile at 6 Mbps

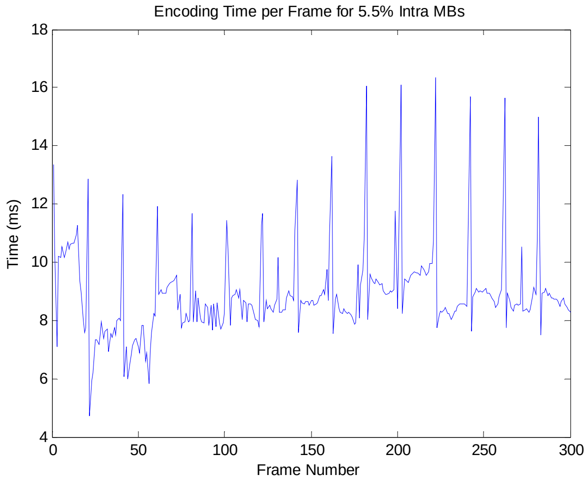

Figure 2.6: Frame Encoding Latency for Intra Period 20, Akiyo at 150 Kbps.

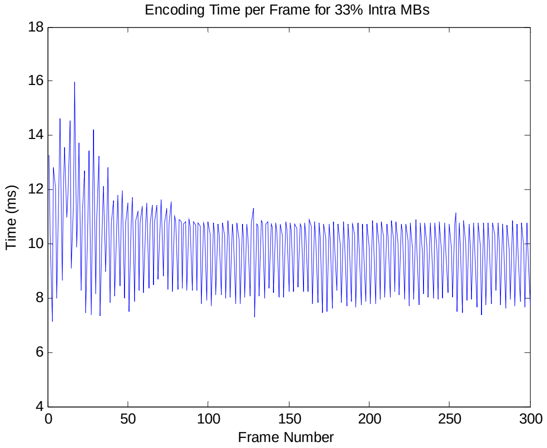

Figure 2.7: Frame Encoding Latency for Intra Period 3, Akiyo at 150 Kbps.

% Intra MB	Average Latency	Standard Deviation	Max Latency	Min Latency	Max/Min Ratio
5.5%	8.41 ms	0.77 ms	11.54 ms	7.06 ms	1.63
33%	8.94 ms	1.60 ms	12.88 ms	7.00 ms	1.84

Table 2.5: Frame latency statistics for encoding Akiyo at 150 Kbps

In the following section the impact of Intra coded information is analyzed for the case of gradual Intra Refreshment of information, where the amount of Intra coded information is more evenly distributed throughout the video sequence.

2.2.2.2 Gradual Intra Refresh

The following graphs show the impact of the amount of Intra coded information with respect to Predictive coded information on the encoding complexity. Time measurements are given for an MPEG-4 encoder running on a Pentium 4 at 2.8 GHz. The vertical axis in Figure 2.8 represents the total encoding latency for the Akiyo (blue), Foreman (red) and Mobile (black) video sequences encoded at the rates given in Table 2.2 and with values normalized to the highest encoding latency. The horizontal axis represents the percentage of forced Intra Macroblocks in every frame of the sequence. This is done by gradually introducing a fixed percentage of Intra MBs in every P frame (Figure 2.8). In particular in our case this is done inserting full rows of Macroblocks following a random pattern but making sure that within the update period all rows of Macroblocks have been intra updated once. Only the first video frame is coded as Intra while all the rest are P frames. This way, a global percentage of 11% Intra MB per frame corresponds to an insertion of an Intra frame every 10 frames, as we could see from Table 2.1.

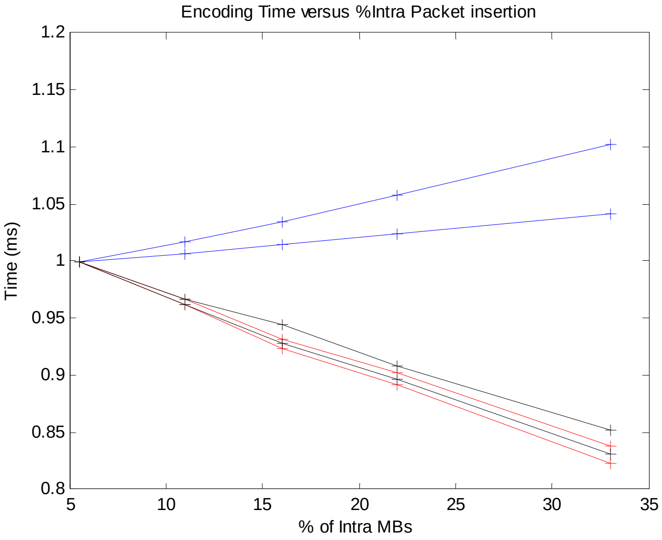

Figure 2.8: Encoding latency versus Gradual Intra refreshment.

We can observe in Figure 2.8 that, for Foreman and Mobile (at different target bit rates, 1 and 1.5 Mbps Foreman, 2 and 6 Mbps Mobile), the higher the percentage of Intra MBs per frame the lower the encoding latency is, achieving for 33% of Intra MBs speed ups of almost 20% for gradual Intra update (corresponding to 1/3 of each P frame coded with Intra MB). This way, we can see how the insertion of Intra MBs could be used as a way to speed up the encoding process to be able to meet more strict timing constraints.

As for the case of Intra Frame insertion, for the low motion Akiyo sequence we can see the opposite effect. As the percentage of introduced Intra MB increases, the latency also slightly increases, 5% for Akiyo at 400 Kbps and 10% for Akiyo at 150 Kbps.

Figure 2.9 shows the impact of a higher percentage of Intra coded information in the individual latency of every frame (instantaneous latency versus the global encoding latency). The encoding latency per frame for the Foreman sequence and two different percentages of Intra MBs: 5% and 33% is compared. Figure 2.11 to Figure 2.14 show the corresponding results of Mobile and Akiyo sequence.

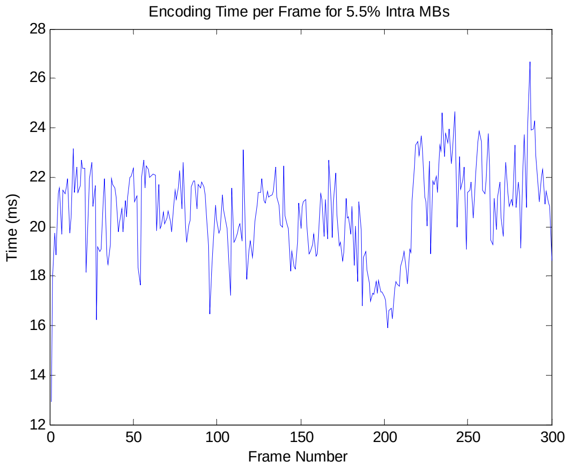

Figure 2.9: Foreman at 1Mbps, 5.5% of Gradual Intra Refreshment.

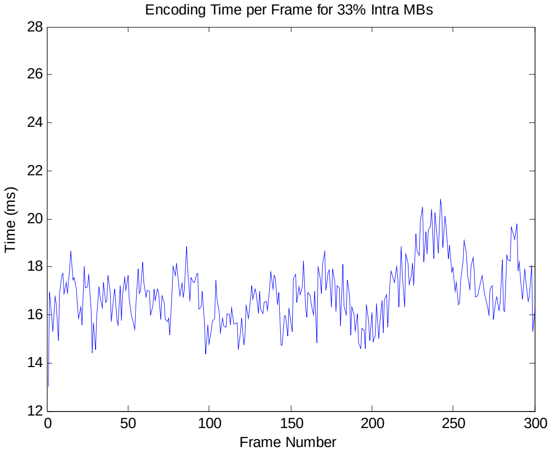

Figure 2.10: Foreman at 1Mbps, 33% of Gradual Intra Refreshment.

% Intra MB	Average Latency	Standard Deviation	Max Latency	Min Latency	Max/Min Ratio
5.5%	20.59 ms	1.83 ms	26.65 ms	12.93 ms	2.06
33%	16.9 ms	1.24 ms	20.81 ms	13.0 ms	1.6

Table 2.6: Frame latency statistics for encoding Foreman

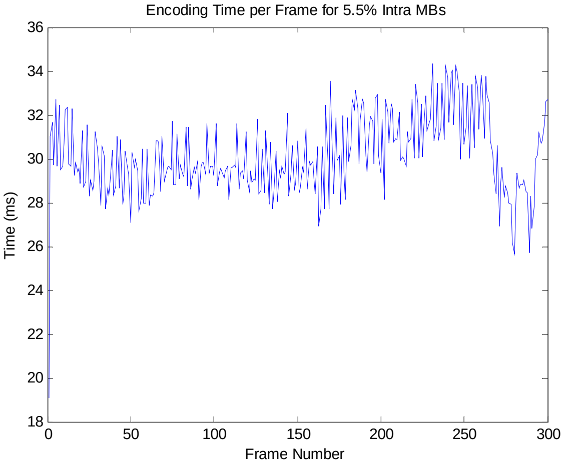

Figure 2.11: Mobile sequence at 6Mbps, 5.5% of Gradual Intra Refreshment.

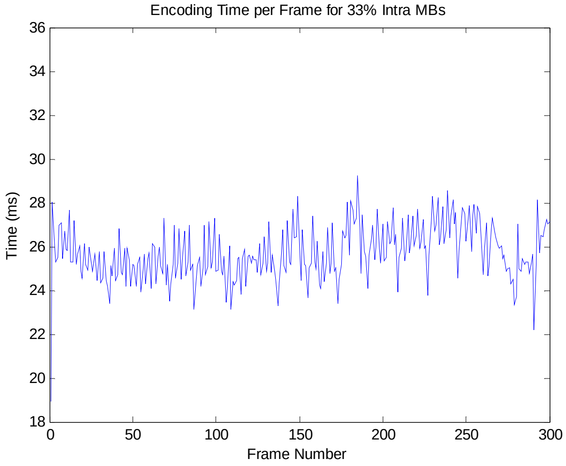

Figure 2.12 : Mobile sequence at 6 Mbps, 33% of Gradual Intra Refreshment

% Intra MB	Average Latency	Standard Deviation	Max Latency	Min Latency	Max/Min Ratio
5.5%	30.12 ms	1.82 ms	34.35 ms	19.05 ms	1.8
33%	25.69 ms	1.24 ms	29.22 ms	18.9 ms	1.54

Table 2.7: Frame latency statistics of Mobile sequence at Rate = 6 Mbps

% Intra MB	Average Latency	Standard Deviation	Max Latency	Min Latency	Max/Min Ratio
5.5%	25.55 ms	1.41 ms	29.37 ms	16.6 ms	1.77
33%	21.29 ms	0.9 ms	24.47 ms	16.66 ms	1.46

Table 2.8: Frame latency statistics of Mobile at 2 Mbps

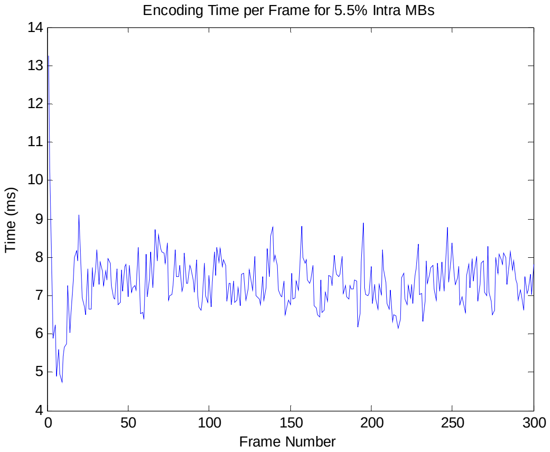

Figure 2.13: Akiyo at 150 Kbps, 5.5% of Gradual Intra Refreshment.

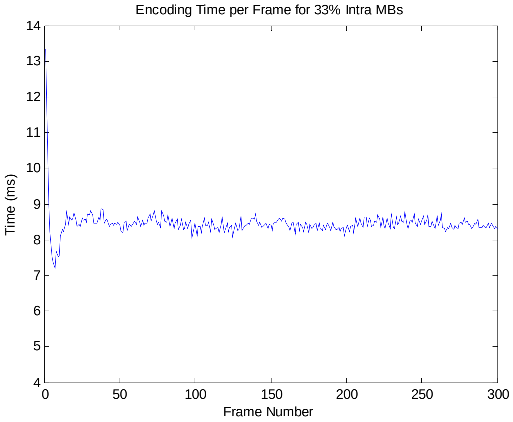

Figure 2.14: Akiyo at 150 Kbps, 33% of Gradual Intra Refreshment.

% Intra MB	Average Latency	Standard Deviation	Max Latency	Min Latency	Max/Min Ratio
5.5%	7.33 ms	0.74 ms	13.24 ms	4.73 ms	2.8
33%	8.44 ms	0.39 ms	13.35 ms	7.19 ms	1.85

Table 2.9: Frame latency statistics for encoding Akiyo at 150 Kbps

For the Foreman and Mobile sequences the latency for every frame with 33% MBs is lower than if it is coded with 5% of MBs. Additionally, we can observe that not only, all individual frame latencies are reduced, but also the latency variations between frames are smoothed down. Table 2.7 and 2.8 show as well that the average and standard deviation values of the frame encoding latency are reduced for a higher Intra MB percentage. From the analysis of

Akiyo in Table 2.9 we see that although the average latency is slightly increased, the standard deviation is highly reduced so as in Mobile and Foreman.

The results show that the encoding latency behaves more constantly and deterministically when a higher percentage of Intra MBs is introduced per frame. This can make the packet delivery to the network more predictable, and depending on the network behavior the packet arrival at the decoder may also have a more predictable pattern. This higher predictability can help the process of task scheduling at both network and decoding devices.

Therefore, using the percentage of IntraMB update as well as the Intra Frame period as breeze description parameters provides with relevant information about the encoding latency variability and the encoder output pattern.

From the analysis, we can also conclude that gradual Intra MB insertion is more suitable than Intra Frame insertion in order to reduce the encoding latency while smoothing the latency variation between frames.

On one hand, the advantage of introducing a higher percentage of Intra MBs (for both Intra frame insertion and gradual refresh approaches) is that the encoding latency is reduced, and the error resilience is increased. On the other hand, the Intra coded information has a negative impact on either the bit rate or the encoded quality. For a fixed target bit rate a higher percentage of Intra coded information causes the error free quality to decrease, as Intra coded MBs are less efficiently compressed. For a fixed encoded quality (fixed quantization parameter) a higher percentage of Intra MBs increases the video bit rate noticeably, around 50% of bitrate overhead for 30% Intra information, as presented in the Quality-Bit rate modeling of Deliverable D2b [6]. This bit rate overhead increases the demands on the network possibly causing the transmission latency to increase and the loss probability coming from overload to increase.

2.2.3 Impact of other semantics: Resynchronization video packet and Data Partition

The use of other tools, such as resynchronization video packets [3] or Data Partition (in which motion and texture information of a resynchronization video packet are separated [3]) has no impact on the encoding or decoding latency. However, the smaller the resynchronization packets are (in terms of bytes, or of number of MBs), the smaller the encoding and decoding latency per packet is. The lower layers could then receive smaller chunks of video data (video packets) at a more regular rate.

3 Decoding Timing Constraints

In this section we analyze the impact of the MPEG-4 semantics on the decoding timing constraints, performing an analysis similar to the one done for the MPEG-4 Encoder. Finally, we address mechanism to speed up the decoder when this cannot meet the timing constraints.

3.1 Impact of MPEG-4 Semantics on Decoding Timing Constraints

This first subsection performs the same analysis of the impact of the proportion of Intra and Inter information, but this time we consider the decoding latency. As in the previous section, we analyze the introduction of intra coded information, by means of Intra frame insertion or by means of an even distribution of Intra Macroblocks per frame.

3.1.1 Intra Frame Insertion

Figure 3.1 analyses the impact of the amount of Intra coded information on the decoding latency when Intra frame insertion has been used at the decoder. Decoding latency values for the Akiyo (blue), Foreman (red) and Mobile (black) sequences at different rates are normalized. The findings are similar to those of the encoding latency. A larger number of Intra frames reduces also the decoding latency for Foreman and Mobile, where close to a 10% latency reduction can be achieved, while it highly increases for the Akiyo sequence.

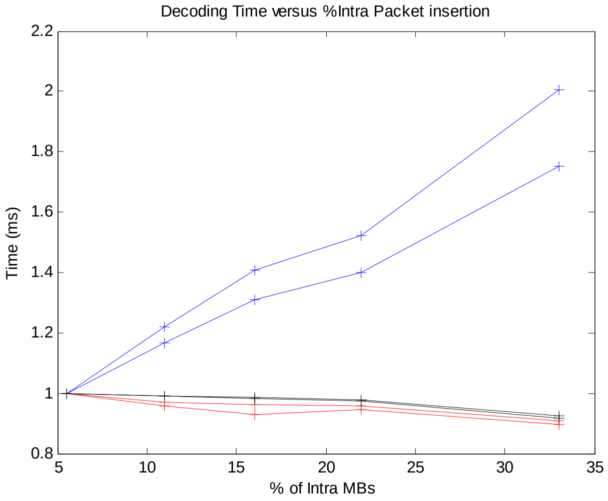

Figure 3.1: Normalized Decoding latency vs % Intra information

Figure 3.2 and Figure 3.3 show the decoding latency per frame for Foreman. For the decoding process it is not so easy to identify in the mentioned figures when the decoding of an Intra frame takes place. As for the encoder, the more Intra frames are inserted the higher the latency oscillation becomes. However, this standard deviation from the average value increases in a smaller degree for the decoding process than for the encoder [See Table 3.1]. This can indicate again that the sensitivity of the encoder with respect to the Intra/Inter proportion is higher than at the decoder.

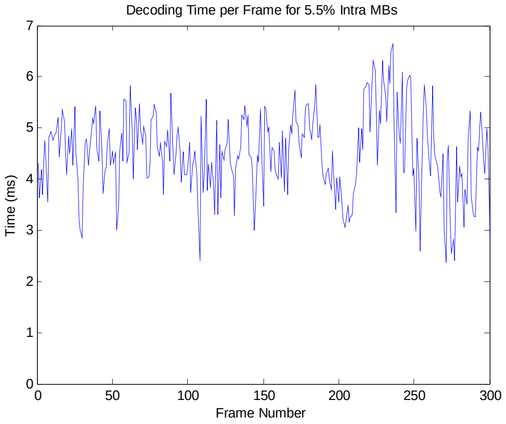

Figure 3.2: Frame decoding latency for 5.5% Intra, Foreman sequence

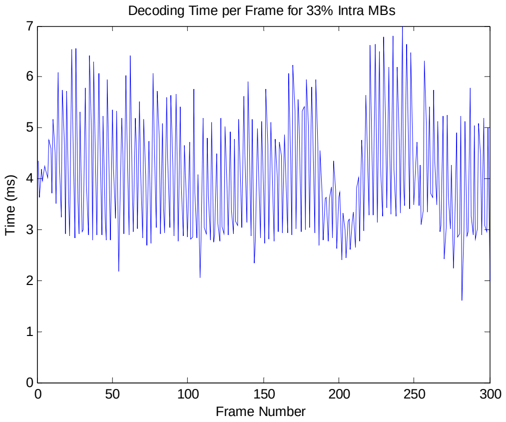

Figure 3.3: Frame decoding latency for 33% Intra, Foreman sequence

% Intra MB	Average Latency	Standard Deviation	Max Latency	Min Latency	Max/Min Ratio
5.5%	4.51 ms	0.80 ms	6.63 ms	2.36 ms	2.8
33%	4.08 ms	1.18 ms	7.04 ms	1.60 ms	4.4

Table 3.1: Frame latency statistics of Foreman

Figures 3.4, 3.5 and Tables 3.2 and 3.3 provide the results for the Mobile and Akiyo sequences. For all sequences, the latency variability increases with the introduction of Intra MB via intra frame insertion. As in the encoding case, the decoding latency increases with higher percentages of Intra MB in the Akiyo sequence, but in the decoder case the latency increases by factor 2, much higher than in the encoder case.

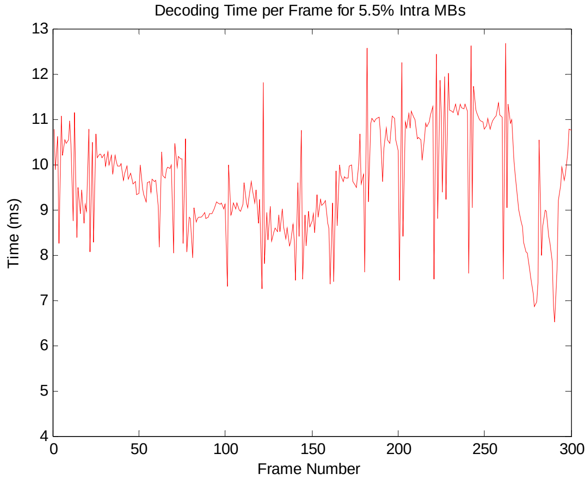

Figure 3.4: Frame decoding latency for 5.5% Intra, Mobile at 6Mbps

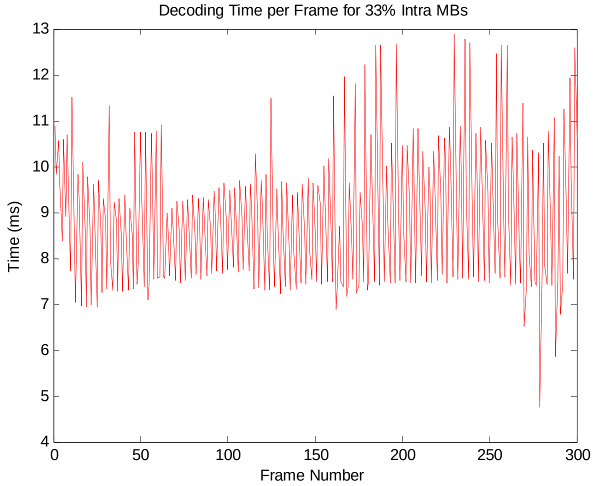

Figure 3.5: Frame decoding latency for 33% Intra, Mobile at 6Mbps

% Intra MB	Average Latency	Standard Deviation	Max Latency	Min Latency	Max/Min Ratio
5.5%	9.63 ms	1.18 ms	12.66 ms	6.5 ms	1.94
33%	8.87 ms	1.45 ms	12.87 ms	4.74 ms	2.71

Table 3.2: Frame latency statistics for Mobile at 6 Mbps

% Intra MB	Average Latency	Standard Deviation	Max Latency	Min Latency	Max/Min Ratio
5.5%	0.86 ms	0.7 ms	4.71 ms	0.22 ms	21.4
33%	1.65 ms	0.84 ms	4.67 ms	0.2 ms	23.35

Table 3.3: Frame latency statistics for decoding Akiyo at 150 Kbps

3.1.2 Gradual Intra Refresh

The following graphs show the impact of the amount of Intra coded information on the decoding complexity when gradual Intra refreshment is used. Figure 3.6 shows a global latency decrease of 15% for 33% Intra MBs for the Foreman and Mobile sequence. The Akiyo sequence, on the contrary suffers a high latency increase (in particular when encoded at 150 Kbps) where the latency is increased by 100%. (See also later in Table 3.7). This is due to the very high temporal correlation in the Akiyo sequence that causes the decoding process of predictive information to be much simpler.

As for the encoding latency, the decoding latency per frame is also reduced and its variability is also slightly smoothed when the bitstream contains a higher percentage of Intra MBs [See Figure 3.6 And Table 3.4]. From this we can conclude, as in the case of the encoding latency, that the percentage of Intra information can be used to reduce the total decoding latency, as well as the individual frame latencies and its variability.

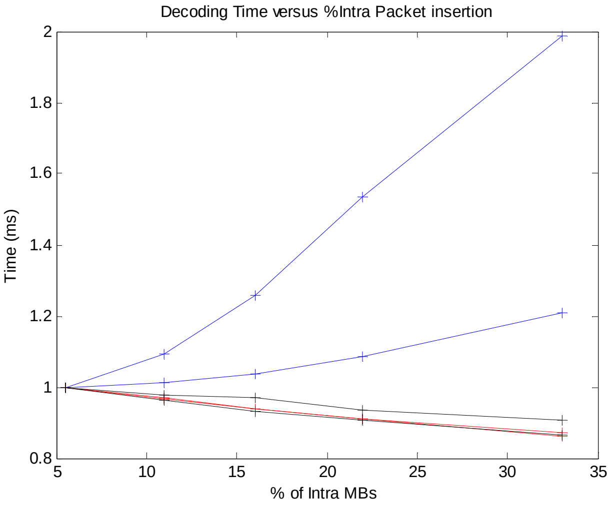

Figure 3.6: Total Decoding latency vs % Intra information

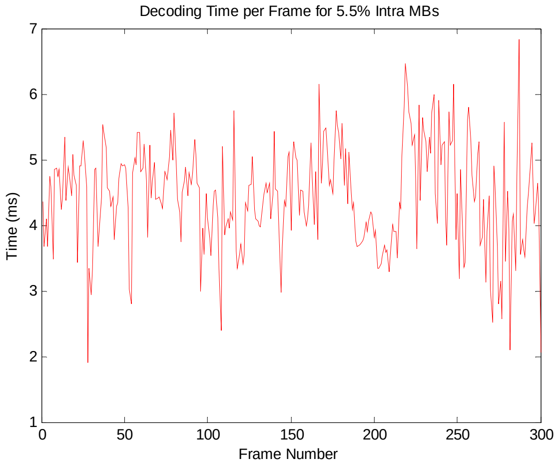

Figure 3.7: Frame Decoding latency, 5.5% Intra, Foreman

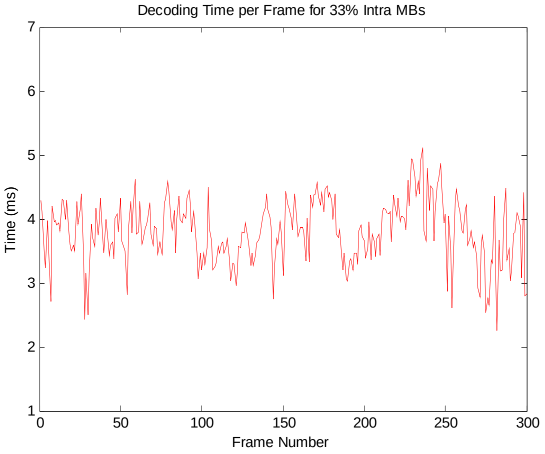

Figure 3.8: Frame Decoding latency, 33% Intra, Foreman

% Intra MB	Average Latency	Standard Deviation	Max Latency	Min Latency	Max/Min Ratio
5.5%	4.42 ms	0.77 ms	6.82 ms	1.90 ms	3.6
33%	3.80 ms	0.49 ms	5.11 ms	2.26 ms	2.2

Table 3.4: Frame latency statistics for Foreman sequence

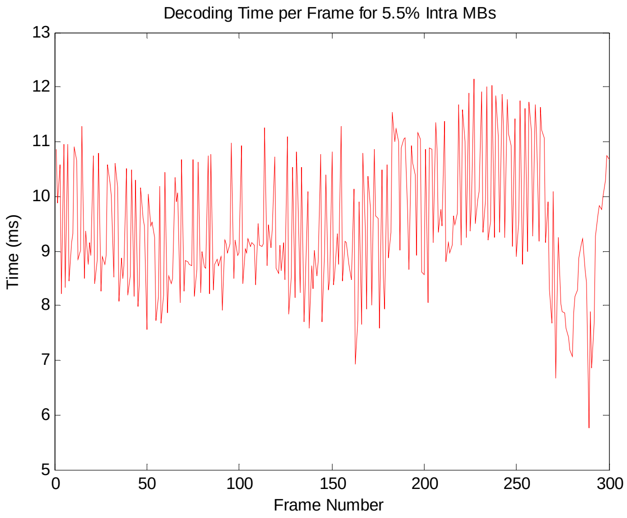

Figure 3.9: Frame decoding latency for 5.5% Intra, Mobile sequence at 6 Mbps

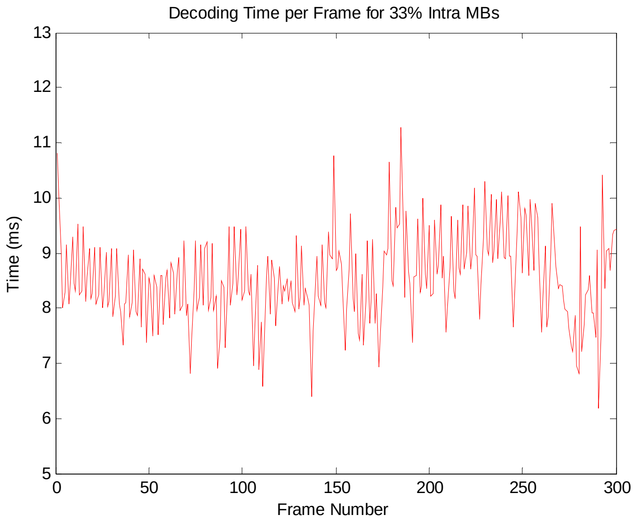

Figure 3.10: Frame decoding latency for 33% Intra, Mobile sequence at 6 Mbps

% Intra MB	Average Latency	Standard Deviation	Max Latency	Min Latency	Max/Min Ratio
5.5%	9.42 ms	1.2 ms	12.14 ms	5.75 ms	2.11
33%	8.54 ms	0.8 ms	11.28 ms	6.17 ms	1.82

Table 3.5: Frame latency statistics for Mobile sequence at 6Mbps

% Intra MB	Average Latency	Standard Deviation	Max Latency	Min Latency	Max/Min Ratio
5.5%	6.15 ms	0.69 ms	8.29 ms	3.86 ms	2.14
33%	5.35 ms	0.51 ms	8.28 ms	4.01 ms	2.06

Table 3.6: Frame latency statistics of Mobile sequence at 2Mbps

% Intra MB	Average Latency	Standard Deviation	Max Latency	Min Latency	Max/Min Ratio
5.5%	0.68 ms	0.29 ms	4.67 ms	0.3 ms	15.56
33%	1.36 ms	0.21 ms	4.68 ms	1.09 ms	4.29

Table 3.7: Frame latency statistics for decoding Akiyo at 150 Kbps

3.1.3 Encoding versus Decoding sensibility to Intra information

From the analysis of the results presented from Figure 2.1 to Figure 3.10 and Table 2.1 to Table 3.7 we can observe the following:

The average frame decoding latency, for this particular implementation, is between factor 4 and 5 smaller than the encoding latency for medium to high motion sequences and up to factor 8 smaller for the low motion Akiyo. The impact of the percentage of Intra information is slightly higher on the total encoding than in the total decoding latency for medium and high motion sequences (Foreman, Mobile) but for very low motion sequences such as Akiyo, a higher percentage of Intra information causes a big increase of the decoding latency.

As for the frame latency statistics the Decoder shows always a higher standard deviation and higher variability between maximum and minimum latency values (see Table 2.1 to Table 3.7), which makes its prediction more difficult and worst case scheduling over dimensioned. A higher percentage of Intra coded information has a bigger impact on the decoder latency statistics than on the encoder.

When the Intra Frame insertion is used, it is easily observable at the encoding side that an Intra frame will lower the encoding latency. At the decoder side, the decoding of an Intra frame may incur in higher latency than a P frame. For both encoder and decoder a bigger percentage of Intra information causes the standard deviation of the frame latency to rise significantly. In addition, the max to min ration increases, mostly for the decoder, where it goes from factor 2.8 to factor 4.4.

When the gradual refreshment is used the variability of both frame encoding and frame decoding latencies is reduced by inserting more Intra information. The frame encoding latency variability is smaller at the encoder side, in particular when a low percentage of Intra MBs is used. When a higher percentage of Intra MBs is introduced, the decrease of latency variability much more noticeable in the decoder case (for instance in Foreman max/min ratio goes from factor 3.6 to factor 2.2 while only from factor 2.06 to 1.6 in encoder).

3.1.4 Comparison with Advanced Video Codec (AVC)/H.264

The analysis performed in this Deliverable is specific to the proprietary MPEG-4 Simple Profile codec [5]. The results are likely to be impacted by the use of another codec such as AVC. A study on AVC was done in [10], the code analyzed is the reference software code [11] (unoptimized code) where the following configuration was studied: IPPP, 7 block sizes, 1 Reference Frame, Search Range of 8, no Hadamard. From the study in [10] a reduction of 25% in the total encoding time was found for both Foreman and Mobile sequences when using a 33.33% of forced intra macroblocks with gradual refresh. This higher latency decrease than the one in MPEG-4 (10 to 15% decrease) may be due to the higher encoding complexity of the AVC in particular for the coding of the predictive information. The use of a high choice of coding block sizes for the motion estimation causes the complexity in the AVC codec to increase with respect to the Intra coding complexity.

As for the impact on the decoding complexity the analysis in [10] shows a small latency increase (less than 7%) in the decoding time for high percentages of forced intra macroblocks (33%), due to the higher cost of the inverse transform in those cases. This contrasts with the MPEG-4 analysis, where the decoding latency slightly decreases for Foreman and Mobile. The reason might be that as the AVC achieves a higher compression degree for the same quantization degree (QP) than MPEG-4 the Rate Control mechanism in MPEG-4 is forced to increase the QP more when higher % Intra is used than the Rate control mechanism in AVC, leading to a lower latency in MPEG-4.

The Akiyo sequence was not used in the analysis in [10], however we can expect that the low motion characteristics of Akiyo cause also a high bit rate overhead and latency increase when the percentage of Intra MB is increased during AVC coding process.

In general, codecs that use more complex tools for the coding of predictive information, or intra information will impact the relative effect of coding more proportion of Predictive or Intra information. For all codecs, however, the influence of the video content remains and it is likely that similar sequences (such as Akiyo with very low motion or Mobile with high motion degree) will have similar trends for different codecs, being the inter sequence variability possibly even higher than the inter codec variability.

3.2 Meeting Decoding Timing Constraints

Decoding MPEG-4 video streams imposes hard real time constraints for consumer electronic devices. The encoded content has to be decoded and played out. Decoding can be performed in hardware or in software, or in a mix of both. Both dedicated and programmable decoders can be based on average-case requirements if they provide means to gracefully handle overload situations. If not, both must support worst-case requirements. However, in a software implementation, it is possible to use the slack on the processor for other applications in average case. With dedicated hardware, there are no such possibilities. As a consequence, the behavior of a software decoder will be less regular than that of a dedicated hardware decoder. If the processor cannot work fast enough to decode all the frames, the decoder has to speed up. One way to do this is by applying frame skipping, this is, not all frames are decoded and displayed, i.e., some of the frames are skipped. The semantics of the MPEG-4 stream play an important role in the skipping policy applied when the timing constraints cannot be met at the decoder.

A controller can be used to decide whether to skip a frame and which frame to skip. In [7] a controller is described that measures the actual finishing time of a frame and the deadline. Once the decoding continues beyond the deadline, the choice is made to stop decoding, skip the frame, or jump the deadline by one frame. The choice is supported by a decision table generated with a Markov Decision model. Given the frame type, the actual decoding time, and the general load, a table indicates the preferred action.

3.2.1 MPEG-4 video Processing

In its simplest form, playing out an MPEG video stream requires three activities: input, decoding, and display. These activities are performed by separate tasks, which are separated by input buffer and a set of frame buffers, see Figure 3.11.

Figure 3.11: MPEG tasks and buffers

The input task directly responds to the incoming stream. It places an encoded video stream into the input buffer. In the simple case, the input activity is very regular, and only determined by the fixed bit rate. In a more general case, the input may be of a more bursty character due to an irregular source such as the Internet or a wireless network. The decoding task decodes the input data and puts the decoded frames in the frame buffers. If sufficient buffer space is available, it may work asynchronously, spreading the load more evenly over time. Its deadline is determined by the requirements of the display task. If B frames are present in the stream, the decoder performs frame reordering, i.e. the display order differs from the decoding order. This means that the frames are offered to the display task at irregular intervals.

The latency variation allowed is a design decision, based on the maximum allowed end-to-end latency, and the available buffer space. If the processor cannot work fast enough to meet the time constraints, the decoder has to speed up. There are two ways to do this: quality reduction, and frame skipping. Whichever strategy is chosen, we assume that the system organization is such that the display task is never without data to display. This is not difficult to achieve. If a decoded frame does not arrive on time, and the display task has to redisplay the previous frame, this is a deadline miss for the decoder. With the given arrangement deadline misses have a penalty, in the form of a perceived quality reduction. Moreover, since the frame count has to remain consistent, the decoder must skip one frame.

3.2.2 Frame Skipping at the Decoder

Frame skips speed up the decoder, and increase the display latency, like a throttle. Unfortunately, the corrective step is rather coarse grained: the display latency is increased by a complete frame period. If the range of allowable display latencies is not large enough, this may lead to oscillation, in which frame skips and bounces on frame buffer overflow both are very frequent. Frame skipping does not come for free. At the very least, the start of the new frame has to be found and the intermediate data have to be thrown away. There are two forms of frame skips, reactive and preventive. A reactive frame skip is a frame skip at or after a deadline miss to restore the frame count consistency. In case of a deadline miss, there are two options, aborting the late frame, which is probably almost completely decoded, or completing the late frame, and skipping the decoding of a later frame. The effects of an abortion and of a reactive frame skip on the display latency are that in the former case, the display latency stays low, and a next deadline miss is to be expected soon. In the latter case, the display latency is drastically reduced, because the decoder will be blocked due to output buffer overflow. An additional frame buffer would give more freedom, and a more stable system, at the cost of using additional memory. In both cases, we have to make sure that the input buffer is large enough to allow the minimal display latency. A preventive frame skip preventively increases the display latency. Skipping a frame takes a certain time, but much less than decoding it. Instead of rising, which is normal for B frames, the buffer occupancy drops during the frame skipping. The decision to skip preventively is taken at the start of a new frame, and is based on a measurement of the lateness of the decoder.

In the next section we present some criteria for quality aware frame selection upon overload situations.

3.2.3 Quality Aware Frame Selection

As mentioned in the previous section, one way of speeding up decoding upon overload situations, is to skip some frames. However, frame skipping needs appropriate assumptions to be effective. Dropping the wrong frame at the wrong time can result in a noticeable disturbance in the rendered video pictures.

In this section we identify some criteria for frame skipping and with quality aware frame selection when it is not possible to decode all frames in time [8].

3.2.3.1 Criteria for preventive frame skipping

Not all the frames are equally important for the overall video quality. Dropping some of them will result in more degradation than others. Here we identify some criteria to decide the relative importance of frames.

Criterion 1: Frame type. According to this criterion, the I-frame is the most important one in a GOP since all other frames depend on it. If we lose an I-frame, then the decoding of all consecutive frames in the GOP will not be possible. B frames are the least important ones because they are not reference frames. Skipping one B frame will not make any other frame undecodable, while skipping one P frame will cause the loss of all its subsequent frames and the two preceding B frames within the same GOP. If we would apply this criterion only, then we would pull out all B frames first, then P frames and finally the I-frame.

Criterion 2: Frame position in the GOP. This is applied to P frames. Not all P frames are equally important. Skipping a P frame will cause the loss of all its subsequent frames, and the two preceding B frames within the GOP. For instance, skipping the first P frame (P1) would make it impossible to reconstruct the next P frame (P2), as well as all B frames that depends on both P1 and P2. And if we skip P2 then we cannot decode P3 and so on.

In [8] the authors provide several more criteria to select the frames to be dropped at the decoder. These criteria indicate which frames should be drop first to minimize the impact on the end quality. However, the work of [8] is focused on MPEG-2, where B frames are available. As the MPEG-4 Simple Profile has no B frame, the criterion about B frames cannot be applied.

Since the criterion 1 is the strongest one, the I-frame will always get the highest priority, as well as the reference frames in the beginning of the GOP.

3.2.3.2 Gradual Intra Refresh versus Use of Intra frames

However, if gradual Intra refreshment is applied instead of a periodical insertion of Intra frames the above mentioned criteria for frame dropping do not hold any more. As the more relevant Intra information is more evenly spread along the sequence, all frames have a similar impact on the end video quality. As opposed to the Intra frame insertion approach, in the gradual refreshment there are no pure I frames (except for the first frame) and P frames, all frames are P frames with a certain degree of Intra and Predictive coded video information.

In a similar manner, criterion 2 about the frame position in a GOP of N frames is not relevant if gradual refresh is used. In a GOP a I frame is inserted at the beginning and followed by all P frames (P0…PN) without forced Intra information within the P frames, this causes an error in an early P frame (Pn) to propagate through all frames (N-n) till the new I frame is inserted. Consequently, a Pn frame with lower n causes a higher number of following P frames to degrade and has a higher decoding priority at the decoder.

With gradual refresh of X% of Intra information per P frame, all information in a frame is intra updated after N=100/X number of frames, which is equivalent to the insertion of an I-frame after N frames. This assures that losses on any frame, regardless of its position in the video sequence, are going to be completely corrected at most after N frames. Therefore the impact of skipping a frame at the decoder is similar for all frames and no priority must be applied in the skipping policy.

3.3 Sublevel Skipping at the Decoder

In order to speed up the decoding process it may be sufficient to skip the decoding of a portion of a frame instead of skipping the whole frame. So as to discard a portion of a video frame without an impact on the decoding of the rest of the frame we can make use of video slices, also called video packets.

A slice is a group of Macroblocks of a frame, which is encoded and packetized as an independent video packet with a resynchronization marker in its header. This allows that after any missing slice in the frame the decoder finds a resynchronization point to start the decoding of the next slice. Moreover the decoding of each slice is independent from the rest of slices within the frame as no prediction between slices in the same video frame is allowed [3].

In each video frame Intra Macroblocks can be used as a resilience tool, in order to refresh the existing information and stop error propagation coming from previous erroneous frames. The introduction of a percentage of Intra Macroblocks per frame can be done in a random manner or following a pattern (e.g raster scan order). Similarly, the Intra Macroblocks can be distributed over the frame or introduced as lines of consecutive Intra Macroblocks. This way, a P video frame partitioned into slices can contain not only P coded slices but also several fully Intra coded slices (I-slice).

The decoding process can choose then to skip the decoding of video slices instead of skipping full video frames when the decoding deadline cannot be met. In this context, we can think that I-slices should have a higher priority as they can stop error propagation and P-slices should be discarded first.

Figure 3. and Figure 3. show some experimental results that study the impact of non-decoded P or I slices on the end video quality. Figure 3. shows the impact of discarding different slice types when Intra frame insertion has been used, in other words, all intra slices are present in Intra frames that are periodically inserted, while no intra slices are used in the P frames. When

Intra slices of Intra frames are discarded the impact on the end quality is quite noticeable, as the error propagation cannot be stopped till the next Intra frame is introduced.

In Figure 3.12, a fixed number of Intra slices per frame is introduced in the Foreman sequence. We can observe how the impact of non-decoded Intra slices is similar to the impact of non-decoded P slices on the end video quality (Figure 3.13). This is due to the fact that the error propagation of both missed slices is corrected by the Intra video packets of the following frames with equal probability. The implication is that no video packets or frames have a higher priority than others when applying a skipping policy at the decoder. As opposed to the Intra frame insertion approach, in the gradual refreshment there are no pure I frames (except for the first frame) and P frames, all frames are P frames with a certain degree of Intra and Predictive coded video information.

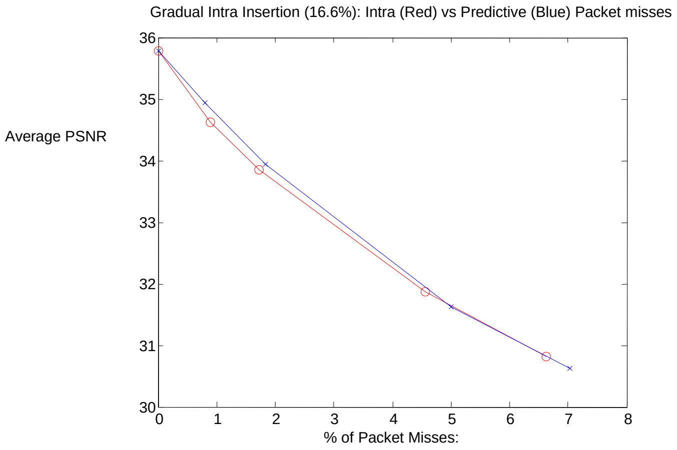

Figure 3.12: Impact of Intra/Inter packet discarding with Gradual Intra Refresh, Foreman

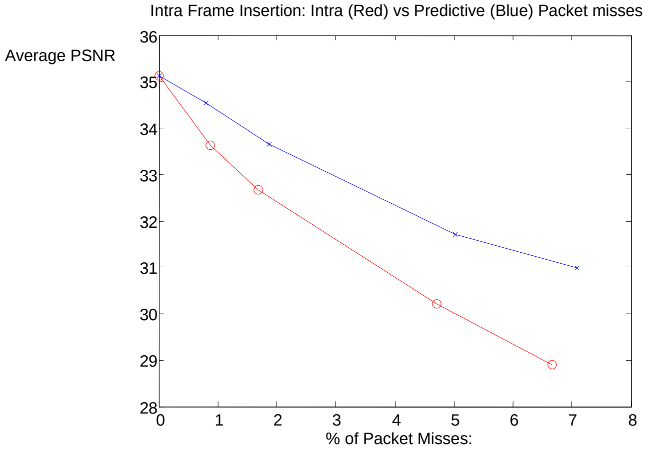

Figure 3.13: Impact of Intra/Inter packet discarding with Intra Frame insertion, Foreman

Figures 3.14 and 3.15 show similar results for the Mobile sequence. As in Foreman, dropping I-packets when Intra frame insertion is used has a bigger impact on the quality than dropping I-packets with gradual Intra refresh but the relative impact is smaller maybe indicating that the concealment mechanism performs better for the Mobile sequence.

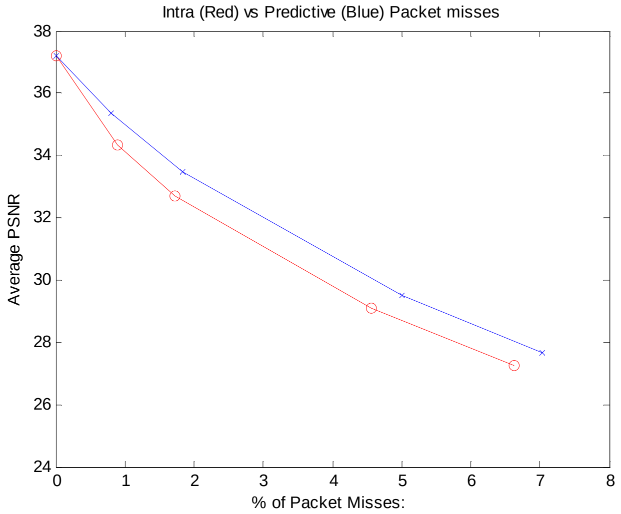

Figure 3.14: Impact of Intra/Inter packet discarding with 16% Gradual Intra Refresh, Mobile

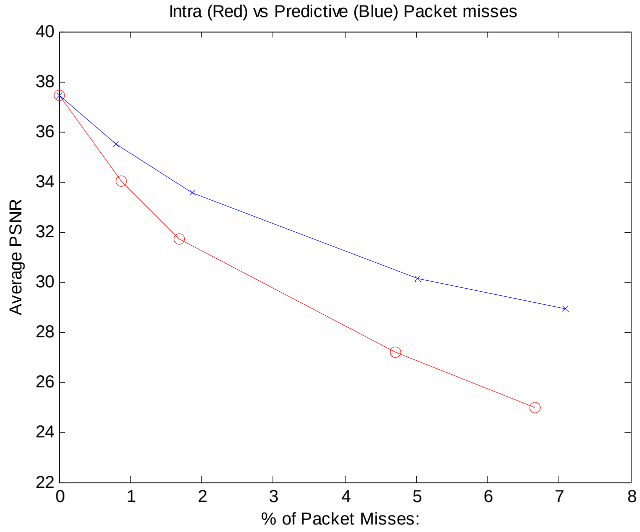

Figure 3.15: Impact of Intra/Inter packet discarding with 16% Intra Frame Insertion, Mobile

3.4 Impact of Data Partition

Data Partitioning enables an encoder to reorganize the coded data within a video packet to reduce the impact of transmission errors. The packet is split into two partitions, the first (immediately after the video packet header) containing coding mode information for each macroblock together with DC coefficients of each block (for Intra macroblocks) or motion vectors (for Inter macroblocks). The remaining data (AC coefficients and DC coefficients of Inter macroblocks) are placed in the second partition following a resynchronisation marker. The information sent in the first partition is considered to be the most important for adequate decoding of the video packet. The video packet in Figure 3.15 shows the way Data Partition is achieved:

- 1st partition: Macroblock number + QP + Motion Data.
- MBM: Motion Boundary Marker.
- 2nd partition: DCT Data.

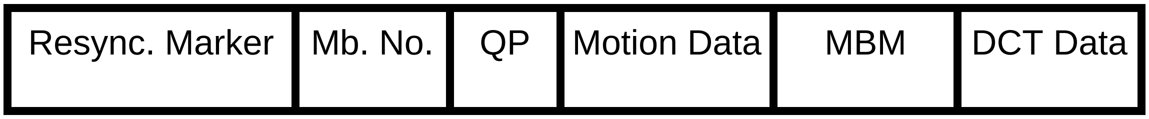

Figure 3.15: Bitstream organization with data partitioning for motion and DCT data.

This way, if the decoding of some information needs to be skipped because the decoding deadline cannot be met, then the Second Partition containing the texture information should be discarded first. The First Partition containing the headers and motion information can still be decoded.

Figure 3.16 show the impact on quality of allowing different percentage of losses (decoding misses) on Texture data. The percentage of misses is given for skipping the partition B decoding of whole video frames.

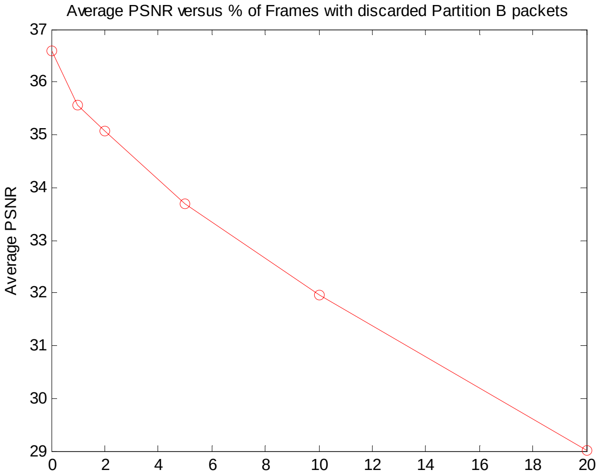

Figure 3.16: Quality versus %Frames with skipped B partitions

Data Partition can be efficiently combined with Unequal Error Protection (UEP) applied on motion or texture [9]. Then motion information, more relevant for the decoding process, can then be delivered with a lower error rate. Moreover, a prioritized delivery of information may also decrease its transmission latency, giving a bigger slack of time for the decoder to perform its task, resulting in less missed deadlines.

3.5 Impact of Decoder Concealment

As a matter of fact, the impact of a frame lost during transmission is similar to the impact of a missed deadline for decoding a frame. In both cases a particular frame cannot be decoded resulting in an impaired video quality at rendering time. To increase this quality, the missed frame is normally concealed using information from previous frames. The easiest way to implement concealment is by replacing the lost information by the previous frame. Other more complex and effective concealment methods can estimate the motion vectors from the lost portion or make use of any kind of spatial interpolations. In any case, the error incurred in the concealed frame will propagate to following frames that use the concealed frame as reference.

In this sense, the better the concealment algorithm at the decoder is, the lower the quality decrease caused by skipping the decoding of a frame will be.

Moreover, the quality of the concealed image depends not only on the concealment method but also on the frame content in terms of motion. If the degree of motion from one frame to another is low, the concealment can be made more efficient.

This motivates that portions of video frames that can be easier concealed should be discarded first than those ones more difficult to conceal that will result into lower end quality. A way to estimate the ease of concealment could be by means of analyzing its motion degree. This could be done at encoding time and passed to the decoder as a kind of metadata signaling which portions are easier to conceal and therefore preferable for being discarded. Another possibility is to estimate the motion at decoding time by for example looking at the size of the first Data Partition, which contains the motion information. A bigger size will indicate a higher motion in the video content.

4 Network Timing Constraints

As mentioned in the previous section, the impact of a frame lost during transmission is similar to the impact of a missed deadline for decoding a frame. In both cases a particular frame cannot be decoded resulting in an impaired video quality at rendering time.

On the other hand, even when video packets are delivered to the decoder successfully, the latency introduced by the network can pose some constraints on the decoding latency. This way, if the video packet is delivered too late to the decoder, this may not have enough processing capabilities to decode the packet and meet the deadline for rendering the information.

Depending on the application we are considering the latency constraints will be more or less strict. For streaming applications, a small playback delay is allowed, which can relax the network latency and make sure the video data is available at the input buffer of the decoder. Other applications, such as real time conversational ones, pose higher latency constraints as no playback delay to absorb the network latency is allowed.

Different parts of the transmitted video information have a different impact on the end video quality. This way, the successful and on time delivery of an Intra frame is more critical than the delivery of a P frame for example as the impairment caused by the loss of an Intra frame is higher. Therefore, the application of content-aware mechanisms that prioritize transmission of critical video information could render better end video quality. The use of Unequal Error Protection (UEP) on Data Partition, where Motion information is more protected than texture information have proven to be effective [9].

Priority scheduling of the most relevant video information can also help increase the end quality. On one hand the relevant information has more chances to arrive successfully (more time available for retransmissions, for instance) and on the other hand its network latency decreases, allowing a bigger slack for the decoding latency.

When congestion or network overload occurs, the sender (or possibly intermediate stations) should throw away frames to control the losses. The loss of a B-frame is preferred over the loss of a P-frame, over the loss of an I-frame. The reception of the frames at the receiver side, and their delays gives information about the bandwidth of the complete transmission chain. Given the bit-rate of the video, the bandwidth of the transmission determines the delay of the frames. Looking at the delay gives information about the required bit-rate. This information can be sent from receiver to sender to remove the less important frames, such that the important ones have a higher probability of arriving in time for the short-term action. The information is also used to adapt the bit-rate of the video to the available bandwidth. The round trip time, which determines the delay of providing information from receiver to sender, needs to be taken into account to calculate the overall delay.

A more elegant method is to use the TCP transport protocol and place an application buffer between application and TCP protocol. When the bandwidth is reduced, the TCP throughput will decrease and the buffer between application and TCP will fill up. At that moment the application can decide to throw away frames selectively. B frames are removed before P or I-frames [12].

4.1 Impact of MPEG-4 Semantics on Timing Constraints

4.1.1 Impact of Intra versus Inter information

As in the case of the encoder and decoder functions, the amount of Intra coded information with respect to the Inter coded information can also have an impact on the network timing constraints.

On one hand, if the rate control mechanism is not used, the insertion of Intra information with a lower compression degree causes the output rate of the encoder to increase. The impact at transmission side will be a higher energy consumption as more bits need to be transmitted plus more difficulties to schedule the transmission of a higher load.

With respect to the communication latency involved in the video transmission, a higher load (encoder output rate) typically increases the network latency. Besides, the risk for congestion becomes higher if in a multi user context the load of the network exceeds the available bandwidth.

Having knowledge at the transmission side of the packet arrival pattern and the statistics of the load demand can be useful for the transmission scheduling process. At the same time it can provide estimates of the energy variation expected at both codec and network sides. In this sense, information about the relative Intra/Inter distribution in the stream can help provide this knowledge at transmission side.

4.1.2 Impact of resynchronization packet size

There is a tradeoff in the selection of the packet size both at the application and link layers. At the application side smaller resynchronization packets allow a higher chance of resynchronization in case of errors but also introduce a higher overhead and reduce the prediction efficiency within a frame. At the MAC layer smaller packets increase the overhead (and with it the transmission energy), see Figure 4.1 from [3] but they also reduce the possibility of a packet being hit by a transmission error, which relaxes the protection requirements.

In our case the selection of the video packet size coincides with the MAC packet size as each video packet is encapsulated in a separate MAC packet, so that a MAC packet loss does not affect the decoding of other independently decodable video packets, which go in other MAC packets.

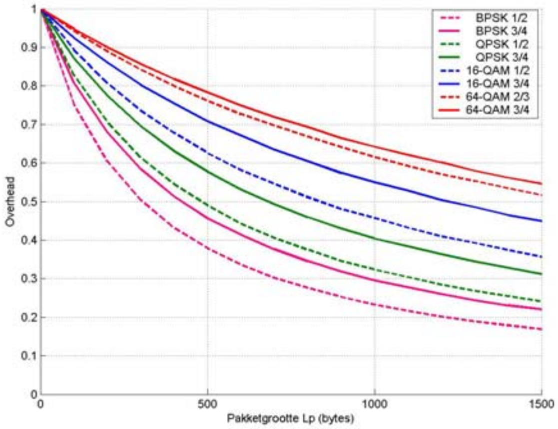

Figure 4.1: Overhead of MAC 802.11a DCF

The general trend is that the higher the bit rate of the encoded sequence (corresponding to lower QPs with a higher encoding accuracy) the higher the energy invested at the network (higher amount of packets to be transmitted). At the same time, for a constant bit rate a higher packet size implies lower energy and latency at the network side, as the overhead is lower. On the other side, when the transmission conditions get worse, big packet sizes have a higher probability of being hit by an error, introducing then more errors at the link level, which decreases in turn the energy efficiency.

On the other hand, there is always a large tradeoff present between energy and latency. To achieve low packet latencies a higher energy needs to be invested at the network, while if we target lower energies the global latency will increase.

The packet header overhead is increased when the number of packets per frame increases. Techniques, which improve error resilience, make sure that a given packet contains only data belonging to one given frame. However the increased robustness has as consequence that some packets are only partially filled and the number of packets per frame increases. RTP packing assures that a packet only contains data of one frame. TCP packing does not. Comparing RTP with TCP packing, we notice that the throughput of RTP is around 20% lower than for TCP. For the more complex environment of Transport Streams the consequences of packet sizes versus the packing of frames into packets is analyzed in [13].

5 Conclusion

This deliverable has presented the impact that the semantics of MPEG-4 have on timing constraints at several functional components of the end-to-end chain: encoder, decoder and transmission functions. We have seen in the first place how the percentage of Intra coded information versus Inter coded information has an impact not only on the total encoding/decoding latencies but also on the per frame latency variability. On one hand, this impact depends on the approach used to introduce the Intra coded information: Intra frame insertion or gradual refreshment. On the other hand, the video content also has an influence as for low motion sequences for instance, the impact of Intra coded information is much higher. This way, the knowledge of the MPEG-4 semantics of a video stream can be used to better predict timing scheduling at the functional components and to prioritize certain information over another. Moreover, the tuning of the parameters at the encoder, can be used to help meet timing constraints at encoding, transmission and decoding components and can help performing a better scheduling.

Secondly, we have addressed the mechanisms used at the decoder when the timing constraints for decoding cannot be met and we have analyzed the impact of the Intra information on the discarding mechanism. Finally, the impact of the semantics on the network timing constraints has also been addressed.

References

[1] Damir Isovic, Gerhard Fohler et al. Report on temporal impact of identified entities on end-to-end-timing, Deliverable D2a, Technical report, IST-004042 project Betsy, August 2005. http://www.hitech-projects.com/euprojects/betsy/results.htm.

[2] Weiping Li, Jens-Rainer Ohm, Mihaela van der Schaar, Hong Jiang, Shipeng Li, “MPEG-4 Video Verification Model version 18.0”, ISO/IEC JTC1/SC29/WG11 N3908 Pisa, January 2001.

[3] C. Blanch, H. de Groot, and P. v.d. Stok. Deliverable D1c: Inventory of MPEG-4 Codecs. Technical report, IST-004042 project Betsy, February 2005. http://www.hitech-projects.com/euprojects/betsy/results.htm.

[4] Information Technology: Coding of Audio-Visual Objects Part 2: Amendment 1: Visual extensions, ISO/IEC JTC 1/SC 29/WG 11 N 3056 Maui, December 1999.

[5] C. De Vleeschouwer and T.Nilsson “Motion estimation for low power video devices”, Proceedings of the IEEE International Conference on Image Processing (ICIP01), Thessaloniki, Greece, October 2001, pp. 953-957.

[6] C. Blanch. Deliverable D2b: Quality – Bit rate modeling without network errors. Technical report, IST-004042 project Betsy, February 2006 http://www.hitech-projects.com/euprojects/betsy/results.htm.

[7] Clemens C. Wüst, Liesbeth Steffens, Reinder J. Bril, and Wim F.J. Verhaegh, “QoS Control Strategies for High-Quality Video Processing”, 16th Euromicro Conference on Real-Time Systems, June 30th – July 2nd, 2004, Catania, Italy.

[8] Damir Isovic, Gerhard Fohler, Liesbeth Steffens, “Real – time issues of MPEG-2 playout in resource constrained systems”, Journal of Embedded Computing, issue 3, June 2004.

[9] S.Worrall, S. Fabri et al, “Prioritisation of Data Partitioned MPEG-4 Video over Mobile Networks”, Packet Video Conference, 2000.

[10] Damian Gonzalez Figueroa, "Error Resilience in H.264/AVC codec", Master Thesis, University of Vigo – IMEC, May 2006.

[11] ITU-T and ISO/IEC JTC-1, 2003. Advanced Video Coding for Generic Audiovisual Services. ITU-T Rec. H.264 and ISO/IEC 14496-10 (MPEG-4 part 10) AVC.

[12] Jeffrey Kang, Harmke de Groot, Peter van der Stok, Dmitri Jarnikov, Iulian Nitescu, and Felix Ogg, *Robust video streaming over wireless in-home networks,* in Dynamic and Robust Streaming in and between Connected Consumer-Electronic Devices, Peter van der Stok editor, pp 193-212, Kluwer, 2005.

[13] Jeffrey Kang, Marek Burza, Peter van der Stok, *Adaptive Streaming of Combined Audio/Video Content over Wireless Networks,* MMNS 2006, Dublin.